THE BEST ... COMPUTER PROJECTS MAGAZINE
MAY 1985

1000 MODEMS TO BE WON

ELECTRONICS & COMPUTING

AN EMAP PUBLICATION

USA $2.95
Germany D6.00
Singapore S$4.95
95p

THE REPAIR RACKET!

HOW TO AVOID THE COWBOYS

HACKERS CORNER
News & the numbers to dial

68 COMPUTING-QL FUNCTION KEY UTILITY

BBC OWNERS-INCREASE YOUR WORD POWER

DIY PLOTTER·SPECTRUM BREAK KEY VECTOR

COVER PHOTO BY ROB BRIMSON

ELECTRONICS & COMPUTING Contents

Vol. 5 Issue 5

PROJECTS

FEATURES

REVIEWS

PLUS

See page 16 for details of the second round of our Hackers' challenge competition and page 33 to be in with a chance of winning one of a thousand modems that are up for grabs in a free to enter competition.

Electronics & Computing Monthly
Priory Court, 30-32 Farringdon Lane,
London, EC1R 3AU

Editorial 01-251-6222
Editor Gary Evans
Deputy Editor William Owen
Production Editor Liz Gregory

Advertising 01-251-6222
Advertisement Manager Tony Herman
Advertising Executive Tracy Keighley
Advertising Production Serena Hadley

Production 01-251-6222
Art Editor Jeremy Webb
Make-up Time Graphics
Publisher Terry Pratt

Distribution
EMAP National Publications

Published by
EMAP Business and Computer Publications

Printed by
Riverside Press, England

Subscriptions and Back Issues
Subscriptions please telephone 01-251 6222 for details.
Back Issues telephone 0858 34567

Electronics & Computing Monthly is normally published on the 13th day of each month.

What good a C5 on C4?

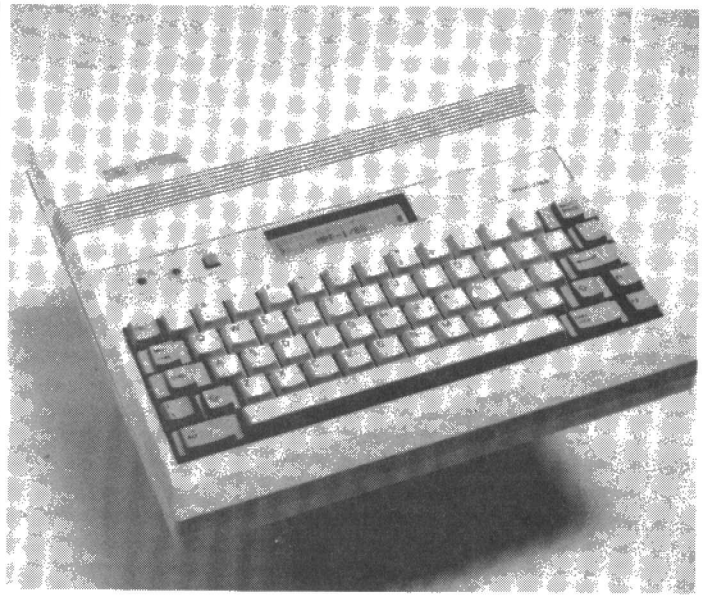
The first series of the Channel 4 programme, 4 Computer Buffs, is now at an end. We commented on the first programme in the run last month, at that time suggesting that the series may well turn into a gigantic plug for a certain well known computer magazine. These fears turned out to be unwarranted and the plug per minute factor was very much less in subsequent episodes. Now that the series is over we are in a position to assess just how well the mass medium of television has served the micro computer owner.

The series had its good points: the photon software transmissions were an interesting experiment making full use of the opportunities offered by TV. Guy Kewney, the only 'real' computer person on the programme, presented some excellent news/gossip pieces, and it was interesting to see the faces behind some of the big names in the industry. Potentially the most exciting element of the programme as previewed were the 'star' interviews. Television, as a medium of communications, can give its best in well conducted, head to head interviews. Unfortunately the interviewers chosen to conduct these chats with the likes of Sir Clive Sinclair gave the impression that they were reading the questions from an autocue, and indeed they probably were. There were no new insights into the industry and there was no attempt by the interviewer to react to answers given to their questions when, possibly, the interview could have explored interesting new ground.

The programme formed part of Channel 4's educational output and there is little doubt that it had educational elements. Whether or not is was for computer buffs is another question. The series reached, for my money, its lowest point in the last programme of the series. Modem corner, by all accounts a popular part of the programme was given over to a Muppet sketch while Guy Kewney was reduced to pedalling a C5 towing a gigantic pizza across the studio. I know computing is meant to be fun but this sort of slap stick computer humour would seem more at home in Spitting Image than in 4 Computer Buffs.

GARY EVANS

Sir Clive Sinclair will tear his hair when he reads about this one. If Acorn lose the BBC contract it would be going to him. This small ad in Byte for 'California Digital' advertises the 'world famous' Dragon 64, 'manufacturered by the Tano Corp.' (who?) 'under license of the British Broadcasting Company'. And all for $139.



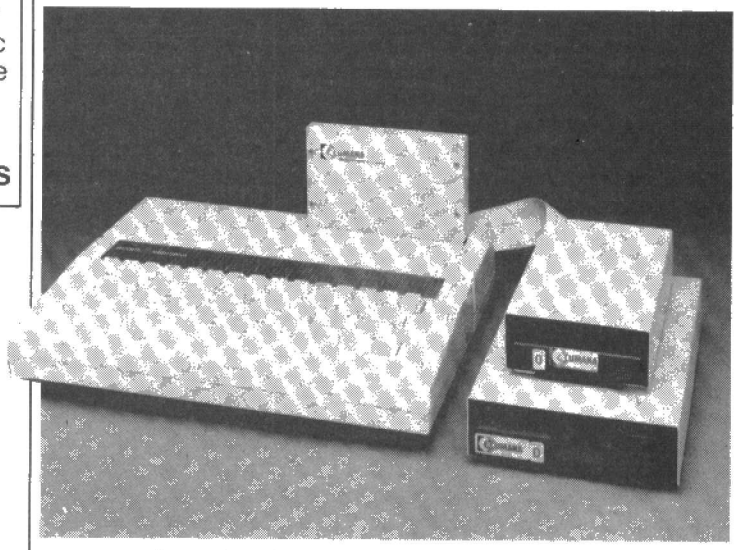
Latest Microprofessor teaches 16-bit computing

The first 16-bit version of the Microprofessor learning system is now available from Flight Electronics. It uses the same processor as the IBM PC – the Intel 8088 – inside an improved package with proper keyboard and liquid crystal display.

The MPF 1/88 is compared by Flight to a Meccano set built about a chip, to enable designers to prototype control devices for robot systems and servo mechanisms, or for use as a single board educational system – this is a computer designed to be opened up and examined.

This is the third Microprofessor development system to be manufactured by the Multitech Industrial Corporation. The first two, single board Z80 and 6502 machines, have already sold over 70,000 units worldwide. Bigger sales are expected for the MPF 1/88 'because of the great interest in the IBM PC and 8088 systems in general'.

Standard hardware includes the Intel 8088 CPU, 4K RAM and 16K



Cumana perks up the ailing Electron with an alternative to Acorn's disk drive interface. The Cumana interface is designed for double density recording and is compatible with both 5.25 and 3.5 inch drives. Up to 89 file names are available on each disk and the interface includes a real time clock/calendar with battery back-up for automatic date stamping of files. The device offers a utility for copying files from the double density Cumana DFS onto a single density Acorn DFS disk that can be used on a BBC micro, an edit program which reads and writes disk directly, a single density formatter which allows disks to be formatted as Acorn DFS disks, and a program to copy files from single density Acorn DFS disks onto double density Cumana DFS disks. *Cumana Ltd, 0483 503121.*

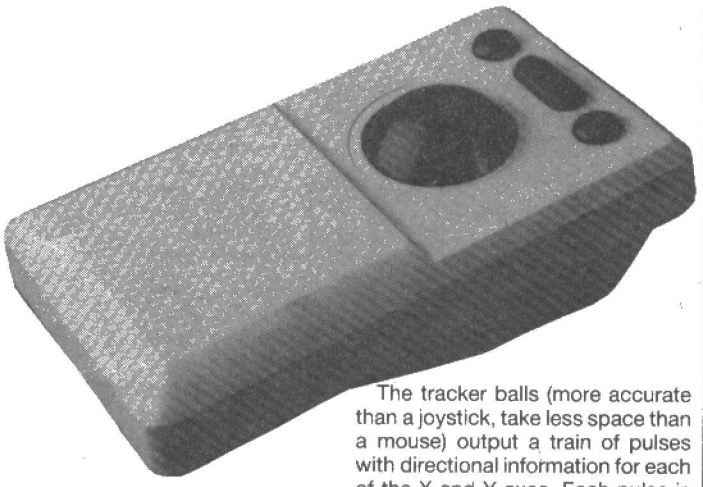
ROM (expandable to 24 and 48K respectively). The keyboard has 59 full-travel keys including two function keys. There is a Centronics interface and a 64 pin expansion bus to allow the MPF 1/88 to interface with many of the expansion cards designed for use with the IBM PC. This facility gives RS232C, memory expansion, video colour output etc.

The Microprofessor comes as standard with a 16 byte monitor featuring many interactive subroutines for Assembly language programming, and support for interfacing to a standard ASCII terminal, a printer driver for Centronics printers, and asynchronous communications routines.

The standard software is intended to teach the user how to write and debug programs, to teach Assembly and the fundamentals of I/O interfacing. The MPF 1/88 is priced at £325.

A wide variety of options will be available with the MPF 1/88 in the first quarter of 1985. Software options include a two-pass assembler with editor for simplifying the assembly language programming process and carrying out complicated programming projects. A ROM-based BASIC interpreter for learning BASIC and a Forth software option will be also available in the second quarter of 1985.

"Because of the great interest in the IBM-PC and 8088-based systems in general, we expect the MPF 1/88 to be an even bigger sales success than our first three systems." explained Max Soffe of Flight.



Tracker Ball for flying aces

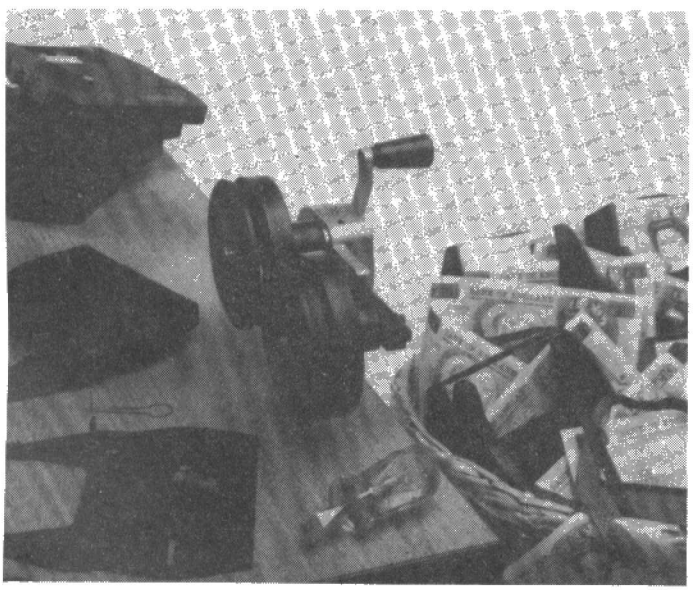
The Marconi tracker ball incorporates many of the features developed by MEDL, one of the Marconi avionics companies, for tracker balls used for military and air traffic control applications.

The tracker balls (more accurate than a joystick, take less space than a mouse) output a train of pulses with directional information for each of the X and Y axes. Each pulse is generated by an optical encoder, and provides an increment of cursor movement enabling precise positional control with zero drift, says Marconi.

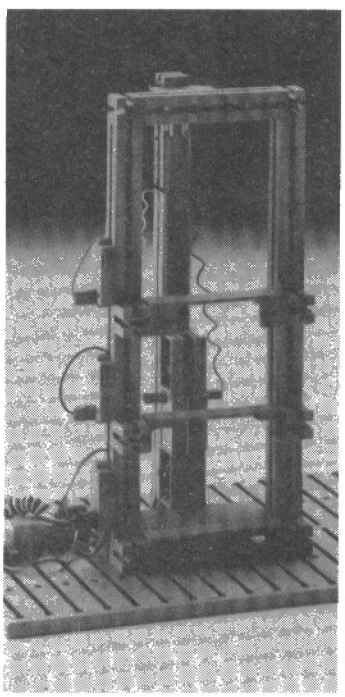
The price of £15.90 (BBC micro and CBM 64) includes graphics software. A roadtest in *E&CM* next month. *Central Trade Exchange 0582 64334.*

Glad to be here

This issue of *E&CM* at some stages during the month seemed doomed not to appear. Half of the staff were struck down by a very nasty flu bug on top of which the Editor's car was broken into and some vital documents stolen. It is only thanks to stirling efforts on behalf of our typesetters, Time Graphics, and our printers, Riverside Press, that we made it to the newsstands on time.



An alternative to *E&CM's* admittedly rather messy scheme for reusing printer ribbons (see March '85 issue) is a service offered by Graphic Display Systems. This company will rewind your used multi-strike printer ribbons for half the list price of the original ribbon cartridge. GDS claims that the second pass of a ribbon through a printer results in little or no loss of print quality – and that a ribbon will often produce acceptab le results after three rewinds. Real skinflints can buy the machine itself for £85, otherwise send your used ribbons to GDS, Bright Haven, Robins's Lane, Colworth, Cambridge CB3 8HH.



No FT – No robot

This computer controlled lift system is one of four new additions to the Fishertechnik robotics kit. There are now sufficient components to build six robot forms – the other new devices include a machine tool, traffic lights ('an elementary introduction to I/O operations') and an aerial rotor with an infinitely variable number of antenna positions, controlled by potentiometers. The kits are available with interfaces for Commodore, Apple II and Acorn computers. *Economatics, 0742 690801.*

LETTERS

Not a nice little earner

Dear Sir,

Thank you for writing about Compunet in your March issue.

However, I would like to correct ' the impression given in the last paragraph that hacking could result in a 'nice little earner'.

Unlike Prestel, knowledge of a user's ID and password is insufficient to get a hacker into Compunet. It would also be necessary to have the user's modem. A unique number in each modem is interrogated on log-in and must tie up with the ID and Password.

In the unlikely event of a user losing both his modem, and ID and Password and not knowing about it, another factor would prevent the Password initialising. Sellers are not paid until the money is collected from the buyer. The buyer gets an itemised statement (again unlike Prestel where only a total of chargeable frames is presented) to check before the bill is paid. They would surely notice and object to multiple purchases of £5 blank frames, especially if their modem and ID had gone missing.

If we come across a potential fraud such as this, we will take action against the seller who is known to us – it is not possible to upload material anonymously.

I hope you will publish this correction – I'm sure you do not wish to be seen encouraging your readers to commit fraud.

Yours sincerely,
J. M. Clarke

Commercial Manager, Compunet

Far from encouraging our readers to comit fraud, the article to which you refer set out to alert organisations such as Compunet to the dangers to the integrity of any data base if security considerations are not given sufficient attention.

While we take the point that the unique number built into a Compunet modem makes the system slightly more secure than other similar services it is by no means necessary for a user to lose their modem if a hacker is to make unauthorised use of their account. The recently published 'Hackers' Handbook' details at least two ways in which such protection may be bypassed.

In conclusion we concede that the 'earner' referred to is not a realistic possibility, there are plenty of schemes that are far more workable – we shall not be publishing these however.

INCREASE YOUR WORD POWER

Mike Williams explains the principles behind a tachistoscope, a device that helps train people who have to make split second decisions, and describes a BBC program based on these ideas.

Look at a picture, or an image on your video screen. As you do so, your eyes will jump from point to point taking in a 'chunk' of the picture each time. While you do this, your brain processes the incoming information and generates some kind of internal representation of the outside world. If you have plenty of time then your eye-brain can make a number of inspections of the picture. But suppose you only get a single flash; how much would you be able to see?

For pilots on the lookout for enemy aircraft one glimpse is maybe all they would get before having to make vital decisions as to whether the plane coming up behind is an enemy or one of their own. Tachistoscopes are devices which are able to flash images onto a screen for a fraction of a second. They are used to train people in situations where split second decisions have to be made.

Most of you are not likely to be interested in training pilots. A course based on the principle of a tachistoscope, using the BBC computer, to help increase reading speed will hopefully be of wider appeal. As you read this sentence your eyes are not moving smoothly across the page but are in fact making a number of jumps. After each jump the eye pauses for a moment while the brain does some processing, and then it moves off again. It seems plausible that if you train yourself to 'see' more at each eye-rest, you should be able to make fewer stops and hence learn to read faster.

The program shown in **Listing 1** is part of a more extensive speed reading course under development. The program is not only fun to use but it could do wonders for your reading speed. It flashes onto the screen some numbers or letters or words, and then challenges you to type in what you saw. If you are correct then you are rewarded with a congratulatory warble and eventually you just might go up a level. That means more numbers, letters, or words, and you will soon discover your 'perception width'. The flash speed is selectable as is the initial level. Try words at fast speed on level nine, after a few beers, for a real challenge!

The trainer offers the choice of using random numbers, letters or words and phrases. You might like to include your own selection of words, for example some simpler words for young children or more technical words. The program could then be useful in

LISTING 1.

```
VDU2,27,64,27,66,2,27,78,6,27,77,8
      ,27,67
      ,27,
100 REM ******************************
110 REM # Speed-Read Tachistoscope #
120 REM *           by             *
130 REM #      M.E.Williams        #
140 REM ******************************
150
160 MODE7
170 :
180 ONERROR GOTO 210
190 PROCinit
200 PROCinstructions
210 PROCmenu
220 IF n_flag THEN PROCnumbers
230 IF w_flag THEN PROCwords
240
250 DEFPROCnumbers
260 PROClevel:size=level
270 REPEAT
280   PROCmake_string
290   CLS
300   PROCshow
310   PROCcheck
320   UNTIL FALSE
330 ENDPROC
340
350 DEFPROCinit
360 ok=0:sn$="":REM string for "under" options
370 VDU23;8202;0;0;0;
380 REM caps lock off for lower case
390 IF ?&216=32 OR ?&216=16 THEN ?&216=48
400 IF ?&602=32 OR ?&602=16 THEN ?&602=48
410 ENDPROC
420 :
430 DEF PROCinstructions
440 PROCheader("Instructions")
450 RESTORE 2270
460 READ A$:REPEAT
470   PROCcenter(A$,VPOS)
480   READ A$
490   UNTIL A$="*"
500 REPEAT:UNTIL FN_c
510 ENDPROC
520 :
530 DEFPROCmake_string
540 Test$=""
550 FOR J%=1 TO size
560   IF n_flag THEN Test$=Test$+STR$(RND(9))
570   IF a_flag THEN Test$=Test$+CHR$(97+RND(26))
580   NEXT
590 ENDPROC
600
610 DEFPROCshow
620 IF n_flag THEN header$=STR$(size)+" digits" ELSE header$=STR$(size)+" characters"
630 PROCheader(header$)
640 PROCcenter(CHR$(132)+"  Press ESCAPE for the menu  "+CHR$(135))
650 PROCwait(200)
660 PROCheader("Press The Space Bar...")
670 PROCBOX(17-size/2,10,22+size/2,8)
680 REPEAT UNTIL INKEY-99
690 *FX19
700 PRINTTest$;
710 PROCwait(speed)
720 CLS:REPEAT UNTIL NOT INKEY-99
730 ENDPROC
740
750 DEFPROCcheck
760 VDU26
770 PROCheader("Enter what you saw...")
780 PROCBOX(17-size/2,10,22+size/2,8)
790 IF n_flag THEN PROCinput(size,48,57) ELSE PROCinput(size,96,123)
800 IF In$=Test$ THEN PROCright ELSE PROCwrong
810 VDU26:REM reset window
820 ENDPROC
830 :
840 DEFPROCright
850 PROCwait(20)
860 ok=ok+1:IF ok=5 AND NOT w_flag THEN size=size+1:ok=0:PROCheader("****** Well done ******"):PROCwait(100)
870 FOR J%=1 TO 10:SOUND 1,-15,10*J%,1:NEXT
880 ENDPROC
890
900 DEFPROCwrong
910 VDU26:REM show correct answer
920 PROCBOX(17-size/2,15,22+size/2,14)
930 PRINTTest$;
940 ok=ok-1:IF ok=-5 AND NOT w_flag THEN size=size-1:ok=0:PROCheader("**** Going down ****"):PROCwait(100)
950 IF size=0 THEN size=1
960 FOR J%=10 TO 1 STEP -1:SOUND 1,-15,10*J%,1:NEXT
970 PROCwait(200)
980 ENDPROC
990
1000 DEFPROCmenu:VDU26,12
1010 ok=0
1020 PROCheader(" Speed-Read Tachistoscope ")
1030 PRINT'':PROClarge("Menu")
1040 RESTORE 1700
1050 REPEAT
1060   PRINT'
1070   READ m$
1080   S$=CHR$(128+RND(6))+A$
1090   PROCcenter(S$,VPOS)
1100   UNTIL A$=" "
1110 PROClarge("Which ")
1120 PRINTTAB(25,VPOS-1);
1130 PROCinput(1,48,53)
1140 PRINTTAB(25,VPOS-1)In$
1150 IF EVAL(In$)=1 THEN n_flag=TRUE ELSE n_flag=FALSE
1160 IF EVAL(In$)=2 THEN a_flag=TRUE ELSE a_flag=FALSE
1170 IF EVAL(In$)=3 THEN w_flag=TRUE ELSE w_flag=FALSE
1180 IF EVAL In$=4 THEN CLS:VDU23;8202;0;0;0;:PROCheader("B,e,...");:PROCwait:200):CALL -4
1190 PROCchoose_speed
1200 CLS
1210 ENDPROC
1220
1230 DEFPROCchoose_speed
1240 CLS
1250 PROCheader("CHOOSE SPEED")
1260 PRINT''
1270 PROClarge("Fast - 1")
1280 PROClarge("Medium - 2")
1290 PROClarge("Slow - 3")
1300 PRINT''
1310 PROClarge("Which ?")
1320 PRINTTAB(25,VPOS-1);
1330 PROCinput(1,47,52)
1340 PRINTTAB(25,VPOS-1)In$
1350 IF EVAL(In$)=1 THEN speed=1
1360 IF EVAL(In$)=2 THEN speed=10
```

improving spelling. If you do want to add words, please note the following points:

(i) The words or phrases must be placed in DATA statements at line 5000, 5010 and so on. This is because of the computed RESTORE at line 1450.

(ii) For the same reason you must not RENUMBER the DATA lines above 5000 unless you modify line 1450 appropriately.

(iii) Each difficulty level up, is a DATA statement with a line number ending in 0, 5040, 5050 etc.

(iv) To keep the listing short only ten different words or phrases have been given for each level. You are less likely to get words repeated if there are more words at each level. If you do increase the number then also modify PROCshuffle by making A$ of greater length and making r% larger also.

The rest of the program is fairly straightforward. Explanatory REMs are included and the PROCs have sensible names. Some of these might well be useful to you in your own programs. PROCinput is particularly useful in this program in that it only allows the keyboard to respond to certain keys. Thus when numbers are called for, only the number keys will respond (other than the dreaded BREAK of course).

So there it is. The program does not guarantee to increase your reading speed by this technique as there may well be other factors determining how fast we read.

LISTING 1 (Continued)

```
1370 IF EVAL(In$)=3 THEN speed=25
1380 ENDPROC
1390
1400 DEFPROCwords
1410 CLS
1420 PROCheader("Word Read")
1430 PROClevel
1440 REPEAT
1450   N%=4990+(10*level)
1460   RESTORE N%
1470   sn=FNget_sn
1480   FOR J%=1 TO sn :REM get one of the words or phrases
1490     READ Test$
1500     NEXT
1510   size=LEN(Test$)
1520   CLS
1530   PROCshow
1540   PROCcheck
1550   IF ok=5 AND level<10 THEN level=level+1:ok=0:PROCheader
"*** Up to level "+STR$(level)+" ***":PROCwait(200)
1560   IF ok=-5 AND level > 1 THEN level=level-1:ok=0:PROCheader
"*** Down to level "+STR$ (level)+" ***":PROCwait(200)
1570   UNTIL FALSE
1580
1590 DEFPROClevel
1600 PROCcenter("Enter the difficulty level",7)
1610 PROCcenter("from 1 to 9.",9)
1620 PRINT''
1630 PROClarge("Which ?")
1640 PRINT'TAB(25,VPOS-1);
1650 PROCinput(1,47,58)
1660 ON INTTAB(25,VPOS-1)In$
1670 level=EVAL(In$)
1680 ENDPROC
1690
1700 DATA"Speed read numbers - 1"
1710 DATA"Speed read letters - 2"
1720 DATA"Speed read words - 3"
1730 DATA"Quit the program - 4"
1740 DATA""
1750
1760 DEFPROCwait(t%):TIME=0:REPEAT UNTIL TIME>t%:ENDPROC
1770
1780 DEFFNy_n:LOCAL I%:REPEAT:I%=INSTR("YyNn",GET$):UNTIL I%
1790 IF I%<3 :TRUE ELSE=FALSE
1800
1810 REM***** MODE 7 procedures *****
1820 DEFPROCcenter(A$,vpos)PRINTTAB((19-LENA$DIV2),vpos);A$:ENDPROC
1830 DEFPROCheader(A$):VDU30,2$;FORI=0 TO 1:PRINTCHR$(&81+CHR$&9D;
CHR$&87;CHR$&8D;:PROCcenter(A$,VPOS):NEXT:ENDPROC
1840 DEFPROClarge(A$):FORI=0 TO 1:PROCcenter(CHR$&8D+A$,VPOS):NEXT:ENDPROC
1850
1860 REM***** General purpose input *****
1870 DEFPROCinput(L%,Lo%,Hi%):REM string length=L% ASC betweenLo% and Hi%
1880 LOCALK%,Z%:K%=0:In$=""
1890 PRINTSTRING$(L%,"."):STRING$(L%,CHR$(8)):*FX15,1
1900 REPEAT:Z%=GET:IFZ%=127ANDK%>0THENPROCdel
1910   IFZ%>Lo%ANDZ%<L%ANDZ%<Hi%THENPROCadd
1920   IFZ%=32 ORZ%=46 THENPROCadd
1930   UNTIL(Z%=13ANDK%>0)OR(K%=L%):ENDPROC
1940 DEFPROCdel:K%=K%-1:In$=LEFT$(In$,K%):PRINTCHR$Z%;" ";CHR$(8);:ENDPROC
1950 DEFPROCadd:K%=K%+1:In$=In$+CHR$(Z%):VDUZ%:ENDPROC
1960
1970 DEFPROCBOX(X1,Y1,X2,Y2)
1980 LOCAL Y
1990 PRINTTAB(X1-1,Y2)CHR$146CHR$234STRING$(X2-X1-1,CHR$163)CHR$181
2000 FOR Y=Y2+1 TO Y1-1
2010   PRINTTAB(X1-1,Y)CHR$146CHR$234CHR$135
2020   PRINTTAB(X2-1,Y)CHR$146CHR$181
2030   NEXT
2040 PRINTTAB(X1-1,Y1)CHR$146CHR$234STRING$(X2-X1-1,CHR$240)CHR$181
2050 VDU26,Y1+2,Y1-1,Y2-2,Y2+1
2060 ENDPROC
2070
2080 DEFFNget_sn:REM Get a new random number
2090 IF LENsn$<2 THEN PROCshuffle
2100 sn=ASC(sn$)
2110 sn$=sn$+MID$(sn$,2)
2120 =sn-64
2130 :
2140 DEFPROCshuffle
2150 LOCALA$,r%,c%
2160 A$="ABCDEFGHIJ"
2170 r%=11
2180 sn$=""
2190 REPEAT
2200   r%=r%-1
2210   c%=RND(r%)
2220   sn$=sn$+MID$(A$,c%,1)
2230   A$=LEFT$(A$,c%-1)+MID$(A$,c%+1)
2240   UNTIL r%=1
2250 ENDPROC
2260 :
2270 DATA," These exercises are to help you", "to increase your 'perception width'"
2280 DATA," Words,numbers, or letters will be"
2290 DATA" flashed onto the screen ever so quickly."
2300 DATA," You then have to type in what you saw."
2310 DATA," You choose the difficulty level," and the 'flash' speed."
2320 DATA," If you do well, you go up a level."
2330 DATA," Press ESCAPE any time to get the Menu."
2340 DATA," Are you ready to start? "
2350 DATA
2360 :
4999 REM do not renumber these data lines!
5000 DATAhave,note,card,beds,tall,belt,shed,boat,sand,your
5010 DATAproper,silent,uncover,slumber,sunbeam,scholar,message,delicious,multiplier,savings
5020 DATA"white house","large fish","fire engine","three boys","little pig","some girls"
"likes them","back door","thank you","fly away"
5030 DATAsentence,flight,report,pepper,phone,plenty,below,clock,rough,voice
5040 DATA"over there","little puppy","please try","good road","that train","rush away","take one","they sang","red apple","can jump"
5050 DATAcommunion,incorporate,enormous,patriot,essential,importance,cascade,fragrant,engagement,tenement
5060 DATA"exactly as he wished","on the opposite side","a quantity of food","well known painter","lettuce was crisp","solve the problem","insists on going","can incorporate","cannot afford it","had an engagement"
5070 DATA"always wore turbans","incessant demands","had an earphone","an imitation pearl","tried to console","was always available","offered an amendment","the puppy whimpered","an electric mower","liked solid food"
5080 DATA"under these conditions","it is just a conjecture","will stipulate the brand","map shows the topography","the younger generation","basic issues involved","prompt reply to a query","transcends all his hopes"
5085 DATA"with excellent retention","is justifiably offended"
5090 DATA"reconstruction period completed","immigration authorities asked","establish a permanent committee","relegated to the fourth division","set of divergent opinions","terminate all previous relations"
5095 DATA"interested in an entire revision","a stretch of undeveloped beach","amazing variety of odd shapes","looking for something lost"
>VDU3.
>*FX3,0
```

U.S. REPORT

Marshal M. Rosenthal reports on a new craze of interactive communication sweeping across the US hacking scene, and on the latest in all-purpose micro music synthesis.

Interactive communication

The pull of interactive gaming is the knowledge that you are not alone as you sit in the dark before a screen: that you are pitted not only against the computer, but also against dozens and possibly hundreds of other games players picking their way through an identical scenario but following no set pattern of play.

Interactive communication, and especially interactive games, are the latest preoccupation of committed computer users in the US. Many of the network information systems offer games where you can go head to head against another person. Some networks offer multiple user games where 10, 20 or 100 or more players can participate at the same time, forming groups and allegiances in simulated battles and adventures.

But the problem has been how to represent the action to the player? While a few systems offer limited graphic symbols on the screen (usually + \ > & and the like), most are text oriented. One exception is the Plato Homelink service.

Plato features outstanding static graphics. The catch was that only IBM PC machines or special (and expensive) terminals could be used. This bypassed all the personal computers. Recently Atari has introduced the Learning Phone cartridge which can access the system. It changes the Atari chameleon-like into an 80 column Plato terminal.

Plato's games combine graphics with a high level of sophistication and intelligence. There are over 25 interactive games on the network. Moria is one of the most popular, and is of the sword and sorcery type. The player establishes a character and then goes out in a land ruled by the laws of fantasy and magic. Some refer to Moria as the world's most confusing and compelling three dimensional electronic maze; it has seemingly endless rooms and corridors. Large numbers can play simultaneously, and a player can continue indefinitely (the game never "ends", although each particular event can be resolved).

Empire is the adventurer's meat and potatoes. 30 people can log-on to the game at one time to form the 4 teams, the Kazari, Federation, Romulans and Orions (this no doubt sounds familiar to fans of Star Trek). Each team controls a portion of space, with each pleyer commanding a spaceship. The purpose is to overun and dominate all of the Universe. A game can last for many hours or several days. Communication is fast and furious, as everyone is making plans and scheming how best to destroy their opponents. Two dimensional graphics are employed in this game of strategy and clear thinking. And of course, a lot of violence.

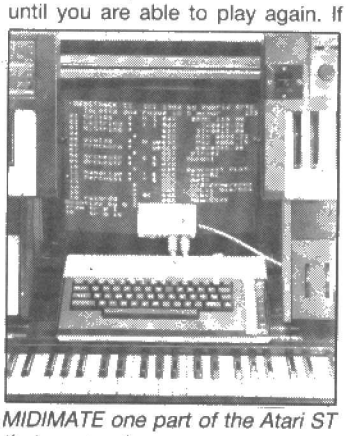
Games Computers Play bases its network on the visual, using Atari graphics to eliminate so-called "user friendly" command menus. The system is accessed with special software which does more than just act as a terminal program. Instead of a log-on code, the screen displays the massive closed doors of the Custom office, which won't open if you're not a member. If granted entry you assume control of a joystick driven robot surrogate. You may now move throughout the multi-coloured city which is composed of a variety of buildings, each relating to a type of user function – like the Post Office for mail and communications. The city scrolls with you as you enter any of the various buildings. The software provided displays the various graphics. Game's computer keeps track of who you are and where you want to go. Entering the Social building allows you to prepare for Spacelanes, the multi-user interactive role playing game. Spacelanes takes place in the Grey Sector, a galaxy of open space and lawlessness. Rules are infrequently obeyed, and fame and fortune await the bold and daring. There are a variety of sentient races, androids (and mutant variations on this theme) that the adventurer must face. He must develop his own resources as he maps his way through space, strange planets and unforseen disasters.

Spacelanes is text oriented, but differs radically in that there is an online referee. The designer, PILOT, monitors each game running, making suggestions and arbitrating events.

Each player assumes a character who can buy (or steal) weapons, aquire wealth, and in general just try to stay alive. The date and time for a game session is posted in advance to give players time to plan and discuss possible scenerios with each other.

Every session is a continuation of the last, and your character is saved until you are able to play again. If you've been killed, well now you can start all over and try to learn from your mistakes. Or make others pay for them.

(See the comms feature on page 23 for more on the use of micros in general communications applications).



MIDIMATE one part of the Atari ST that you can buy now.

The MIDI connection

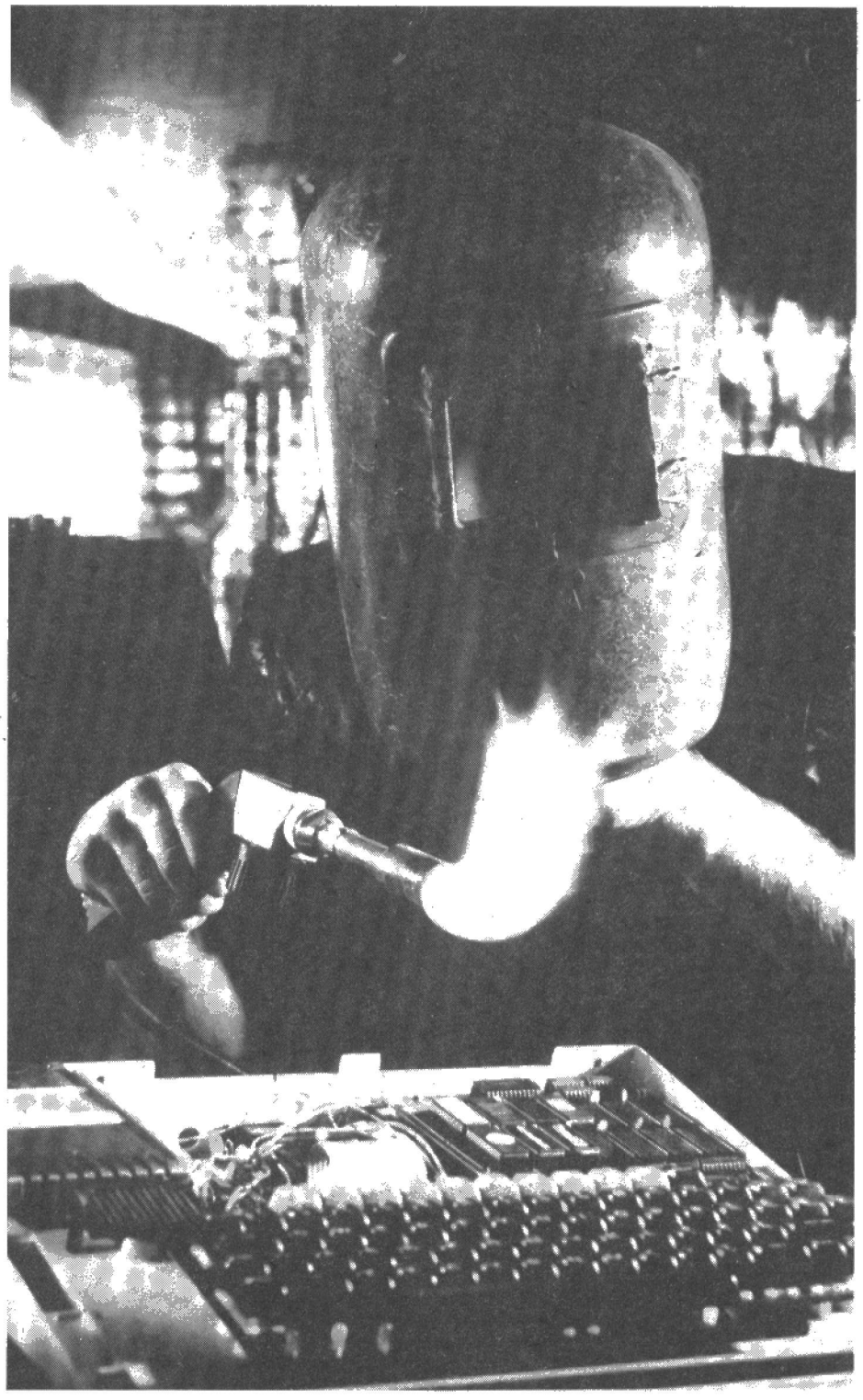
There was a time when a synthesiser or electronic piano cost a fortune. Today you can buy a whole band and pack it in a briefcase. So it would seem natural for computers to have moved into this area a long time ago. Well, they did – the big, powerful ones, not the personal 8-bit machines. Even so, there was a compatibility problem in the musical instruments themselves; they just weren't made to interface with one another.

MIDI (Musical Instrument Digital Interface) is a recent standard which allows various musical keyboard and synthesiser devices to communicate with each other, in much the same way an RS232 port facilitates data transfer between a computer and various peripherals. Hybrid Arts of California has perfected the MIDIMATE; an integrated hardware/software package which works with Atari computers, past and present (a version is installed in the new ST line as well). Using menu driven software, a user can record, compose and play-back multiple tracks of music, taking full advantage of the various strengths and capabilities of up to 16 independent musical devices.

Bob Moore is the president and chief software designer of Hybrid Arts. His young appearance belies the fact that he has been working with electronics and synthesisers for many years. 'The MIDIMATE allows an incredible flexibility,' says Moore. 'The possibilities are incredible.' The software is impressive. Besides visual/aural metronomes, tempo controls, sequence tracks, overdubbing; devices can be remotely controlled, each note precisely monitored and accounted for. Moore notes that the MIDIMATE has been given a tough construction for the novice and musician alike. "It's a sealed unit, you can't hurt it," he adds.

Moore's eyes light up when he mentions MIDICOM, his modem software package. "It allows people to share their music with each other all over the world," he says. "We expect to see some amazing things come out of international communication. After all, musicians don't exist in a vacuum." Other packages include Session Sounds (which contains music developed by studio musicians), and Session Player (prerecorded tracks of bass lines, chord progressions and drum parts).

The alliance with Atari is seen as a way to propel both companies into the limelight of professional applications. "You name it and MIDIMATE can do it," adds Moore.



EXTENDING THIS MORTAL COIL

'A gigantic hiccup looming in the computer revolution' is how one micro repair organisation describes the burgeoning problem of the broken computer.

They may be overstating their case, but the facts remain that there are now well over one million home computers in use out of the warrantee period, and that the controversy over reliability has reached such a pitch that the chairmen of two leading manufacturers came to blows in a Cambridge pub.

Electronics and Computing set out to investigate how big the problem really is, what can go wrong with your micro, and how you go about getting it fixed. We spoke to both manufacturers and third party repair companies about the level of service offered, and standards of quality control in the industry.

Last year Acorn's advertising agency commissioned a survey on micro reliability from Business Decisions Ltd. The survey, which so disturbed Sir Clive Sinclair, claimed that 25.4% of Spectrums, 18% of Vic 20s, 13% of Commodore 64s, 5% of BBC micros and Electrons, and 4% of Amstrads (well done Amstrad) were returned as faulty within six months of purchase. 'A completely silly and skewed result' insisted Sir Clive. He may have been right, in that the period included Christmas when Spectrum sales were very high, and the survey was not commissioned by an uninterested party.

Sinclair claims that 40% of Spectrums returned have no fault and that the true return rate is more like 5%. While our research did not confirm the findings of the Acorn sponsored survey, it did point to certain inherent weaknesses in some machines, occasionally admitted even by the manufacturers.

Take Commodore for example. They recently suffered a number of returns of CBM 64s with faulty disk controller chips. The fault appeared to users as the disk drive going down, and defied test procedures because at no time during quality control, when the CBM 64 is burnt in for three hours, is it linked to a disk drive. Commodore is now investigating the source of the problem, right back to the guilty chip's home in Hong Kong, and is revising its quality control procedure. At the same time Commodore insists that the vast majority of breakdowns (70%) are caused by user error.

Lawrence Hardwick, who looks after Acorn's service operation, claims a high reliability record based on the prompt correction of past errors. The only problem over which he admits to having little control is that of faulty connectors: 'in any piece of equipment, not just the BBC micro, the thing that is most likely to fail is a mechanical part'.

Most service companies emphasise the low failure rate of the BBC micro; it is 'a very good machine for reliability' says Peter Venes of Hemel Hempstead Computer Centre (he adds as an aside that the Dragon's IC15 sound chip is forever blow-

ing up). John Serlin of Suredata Computer Services confirms Lawrence Hardwick's point about occasional poor contacts on the BBC, as does Mark Duffill, of Micro-Fix, who has lately come across a number of BBCs with faulty 74LS245 data buffer chips. 'There used to be a high proportion of faulty ULAs on the BBC, and it is still quite common to find failures in the 6522 system and VIA'. Duffill also relates some horror stories about users who have tried to plug a mains supply into the RGB interface, or disk drive cables into the 1MHz bus. 'Boards can be wrecked by inexperienced users trying to do upgrades or repairs; they often lift up tracks or blow the power supply', but 'Generally, since Issue 7 the BBC has been a reliable machine. Issue 4 machines are also well made but Issue 3 in particular has mistakes on the board and at the cassette interface – upgrades require cutting links and rewiring'.

As usual, nobody agrees about the Spectrum. Most faults relate to the power supply, which can be overloaded by peripherals, especially those which don't come up to spec. This manifests itself as a memory failure. Jim Serlin puts most of the blame on users who try to plug in peripherals while the computer is on, or on badly designed peripherals. 'Occasionally', he says, 'the locating pin is either in the wrong position or falls out, with disastrous results'. The symptoms are as follows: 'When the 16K machine's memory goes down you get a black screen with vertical lines. The 48K computer will initialise but you can't input anything'.

Memory failures can be caused by an internal power supply failure or by a faulty 4116 memory chip – the Spectrum has 16 RAM chips all ready to forget themselves at a moments notice. Plugging in a peripheral while the Spectrum is switched on causes the disappearance of the –5V line which supports the memory, or a complete blow out of the power supply generator. It must be stressed that this isn't a design fault. Lawrence Fritwell of Computer-fix defends the Spectrum and says that 'proportionally to volume and sales there is no apparent difference between one machine and another'. When confronted with the charge that Spectrums are unreliable, Fritwell responds immediately: 'That's unfounded when you consider the number of Sinclair units that are sold'. Computer-fix has a Sinclair franchise to repair Spectrums under warranty.

Getting it fixed

Most micros are sold with a one year warranty. This means that in that period the manufacturer undertakes to make all repairs free of charge, unless the warranty has been invalidated. On most micros you can do that simply by breaking the seal on the case.

Acorn 'would really be worried about obvious signs of physical damage and changes which make our diagnostic tools no longer workable' (eg adding extra ROM or RAM boards). Technically, you will invalidate a Spectrum's warranty by opening the case, but if you are careful to avoid scratching the screws (clingfilm round the screwdriver?) and don't rip out the innards the machine will replaced if a fault appears. I have heard of one individual who inserted a chip of his own devising into a 16K Spectrum, melted down the whole board as a consequence, and on sending the machine to Camberley received a brand new 48K machine by return of post – but this procedure is not recommended. Attaching a new keyboard to a Spectrum will also invalidate the warranty. The cases of some micros have a seal – if it is broken that will automatically invalidate the warranty.

Most manufacturers will service a computer outside its warranty period – for a price – by subcontracting the work in the same way that they subcontract manufacture. Acorn have approved service centres too numerous to mention; they do not guarantee a fixed charge for repair. The company will set up a service centre where none exists for any user of over 100 machines (eg education authorities, the Ministry of Defence, Plessey etc).

Acorn 'would really be worried about obvious signs of physical damage and changes which make our diagnostic tools no longer workable' (eg adding extra ROM or RAM boards). Technically, you will invalidate a Spectrum's warrantry by opening the case, but if you are careful to avoid scratching the screws (clingfilm round the screwdriver?) and don't rip out the innerds, Sinclair policy is (or rather was, see below) to replace the machine once a fault appeared. I have heard of one individual who inserted a chip of his own devising into a 16K Spectrum, melted down the whole board as a consequence, and on sending the machine to Camberley received a brand new 48K machine by return of post – but this procedure is not recommended. Sinclair has recently embarked on an economy drive, and soon only in exceptional cases will a machine be replaced, except within 30 days of purchase. The company has employed a quality controller (was this a direct result of the Business Decisions report?) and clearly a lot more effort is being put into ensuring that machines which leave the factory gates do so in reasonable condition. Incidentally, attaching a new keyboard to a Spectrum will also invalidate the warranty. The cases of some micros have a seal – if it is broken the warranty will be automatically invalidated.

Sinclair subcontract all repair work, inside and outside warranty, to Camberley based Computer-fix. There is a flat fee for any type of repair of £19.55 inclusive of all labour, parts, VAT, postage etc. The com-

Dos and Don'ts

Thje user has no control over faults resulting from manufacture or design, and it is probably true to say that most micros break down because they are mistreated. PCBs are generally reliable and therefore so are most home computers – the only vulnerable parts are the keyboard and connectors. The user does have control over how the computer is treated, and because prevention is better than cure it is worth taking a look at how not to treat a micro before considering the worst – ie breakdown.

The first thing to watch is where you put your computer: these machines dislike heat, knocks, and magnetic, electrical or radio interference. So don't leave a computer in direct sunlight or in any place that errant humans are likely to crash into. Heavy switching equipment, such as a lift, may cause electromagnetic interference or power surges on the mains; this will corrupt your data. The answer is to install a spike suppressor, which will regulate the voltage supplied to your computer. If you are losing data for no apparent reason the culprit may be that fluorescent light overhead – they can emit RF interference.

The power supply is often the source of memory or system breakdown and should be isolated from other electrical equipment, for example the television. Ensure that it is not overloaded by add-on memory or graphics boards or interfaces.

The most common causes of breakdown are also the most obvious, for example, plugging in peripherals or removing or inserting ICs when the power is on – don'-t do it. Another is the removal of disks while they are being accessed by the drive head – don't do it. Even worse is trying to move the computer while it is switched on; this may cause no harm to a Spectrum but with a BBC and disk drives you will be sorry.

Magnetic media are as important to the system as the hardware – and usually more delicate. Tape users should regularly clean the recording head of their machine (the value of cleaning disk drive heads is less certain – you can do more harm than good). Disks should never be bent, or subjected to smoke, dust, magnetism or water vapour, unless you want to lose your data. Smoking near a disk drive (author stubs out cigarette) will result in deposits of carbon on disks and drive head. Blowing on a disk doesn't clean it – the water vapour in your breath will cause damage. Store disks in a purpose built case at the same temperature as the computer, and don't forget the gold rule: ALWAYS MAKE REGULAR BACK-UP COPIES.

pany uses custom-built test equipment called 'Acumen', which it says enables one technician and one solderer to repair up to 50 computers a day. Acumen diagnoses IC faults and, says Computer-fix fatigued components about to fail. It also can be used to tune the video controller. There is a guaranteed turn-around of 48 hours plus time in the postal system.

Computer-fix will also repair Commodore Vic 20 and 64s, the Sharp MZ700, Atari machines, BBC micros, Electrons, Dragons, Orics, and the TI994A. There are fixed charges for the BBC (£40.25) Vic 20 (£25.00) and CBM 64 (£37.50). Disk drives and printers are repaired on a one-off basis. 'Most units receive a 7 day turn-around', says commercial manager Lawrence Fritwell. 'We keep a large number of spares in stock but obviously not all. BBC and Commodore parts occasionally cause some difficulty'.

The Quantum Sinclair Service Centre will repair all Sinclair machines and peripherals (RAM packs, Interface 1 and Microdrives) and some third party peripherals ('the Cheetah devices are more difficult'). There is a fixed labour charge of £10 plus parts: 'the average repair is below £20, some as low as £13'. There is a warranty on labour charges after repair. The company offers one piece of helpful advice to owners of DK'tronics keyboards: 'always remove the power supply from inside the case before sending it through the post, because often it will tear itself from its mountings and wreck the PCB'.

M icro-Fix is a London based company specialising in repair and maintenance of BBC micros and the Acorn Econet system. They will also repair Mitsu, Shugart, Acorn Olivetti, Teac and Cumana drives, Microvitec monitors, and Epson FX80 printers ('they are fairly reliable – we don't get many in'). The company will soon be buying diagnostics for the Acorn second processors and, 'if Solidisc is on board we can check it out too'.

Micro-fix offer on-site as well as off-site maintenance for people in a hurry, and extended warranty maintenance contracts. On-site repairs cost £40 in the Greater London area, £60 outside it. The Econet service includes installation and repair, with a guaranteed response time of

"... Computer breakdown a major problem ..."

48 hours on-site and 72 hours off-site. Three kinds of contract are available: depot repair (6% of equipment list price); on-site repair (10% of list price; and 2nd line maintenance at $15 per hour labour costs plus parts. Services of this kind will be of interest to business users to whom immediate repair is vital to their business interests, and educational establishments.

Insuring against disaster

Insurance is an alternative to one-off repairs and lot cheaper than – but not as comprehensive as – a full maintenance contract with a guaranteed response time for on-site repairs. A wide range of home micros are insured by the Micro Repair Club through the Domestic and General Insurance Co. D&G underwrite the Commodore Cover Plan, Atari Service Contract, and Amstrad Computer Protection Plan, as well as the Computer-fix 'First Aid Plan'. General cover is available for any home computer, monitor and printer for up to five years. These plans cover full cost of

"... DIY modifications will usually invalidate any warranty ..."

repairs out of warranty (parts and labour) or value of the equipment if repairs cost more than the original purchase price. Insurance is also available for fire and theft.

Prices range from £12.60 for two year cover for a computer valued at under £150, to £69.60 for a computer valued between £300 and £600, covered against breakdown, fire, and theft for five years. Cover for disk drives, printers and monitors is extra.

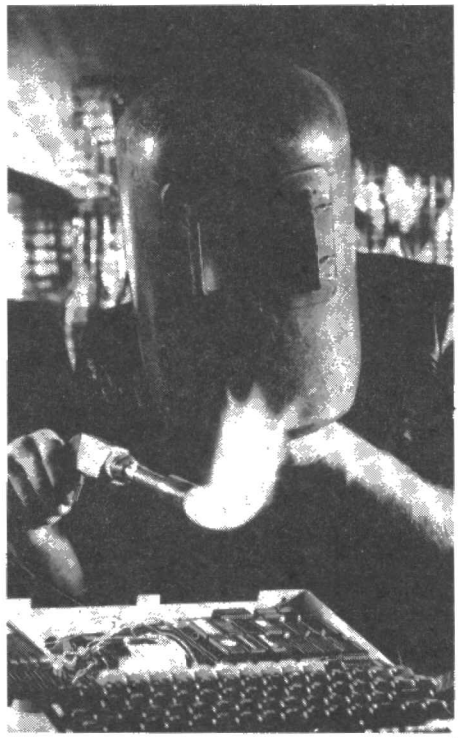
The Commodore, Amstrad and Atari plans are marketed in association with the manufacturer to cover repairs outside the warranty period.

Needless to say it is possible to take out insurance *after* your micro has broken down (as long as you don't tell the insurer!) and then put in an immediate claim. Although the insurance company can do little to stop it, strictly speaking this is fraudulent and not recommended.

The least to expect

Maintenance problems vary immensely. A Spectrum owner with Microdrives and Sinclair thermal printer has very different problems to a BBC micro owner with Z80 second processor, business software, disk drives and printer. The latter should expect the dealer to organise all repairs, at least within a year of purchase. The Spectrum owner will have to make do with the many repair shops springing up to cure 'the gigantic hiccup' but should expect to pay less than £20.00, and to have a guarantee on the repairs done – any secondary faults or poor repairs will show up within three months.

Because of the specialist diagnostic equipment required, few of the smaller repair outfits will service anything other than the most common low-cost machines. For repair of disk drives, printers and older 8-bit machines it may be necessary to go to one of the many business micro repair companies who also do one-off repairs. Extensive repairs will usually be expensive, but of course the only alternative is do-it-yourself. *E&CM* found one company, PC Engineering and Maintenance Ltd, who are prepared to repair 'virtually any 8-bit or 16-bit computer' on an ad-hoc basis (see Table).



Who to contact

Below is a list of companies specialising in home computer or small 8-bit computer repair. The list is not exhaustive and we apologise in advance to anyone who has been left out. Where a phone number is followed by letters in brackets, this indicates that the company repairs one or two computers only, ie S = Spectrum, C = Commodore, B = BBC, Z = ZX81, D = Dragon.

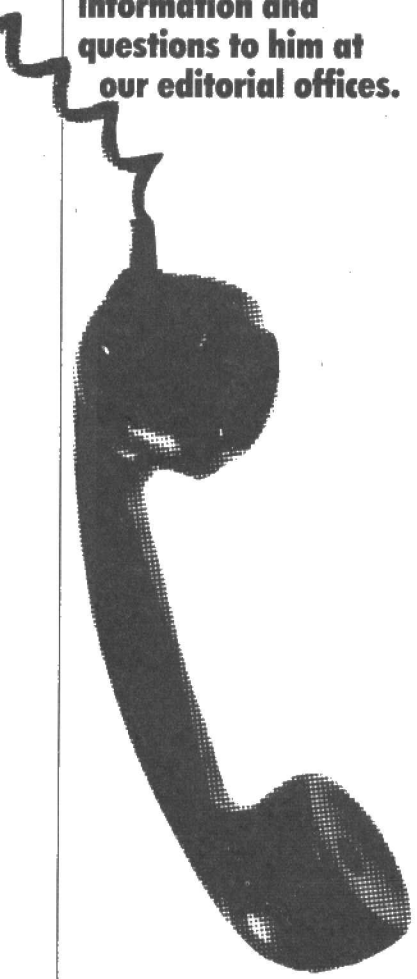
Armsure Ashton-under-Lyne, Lancs. 061 339 8766 (S, Z). **B. C. Bunce & Son** 36 Burlington Road, Burnham, Bucks (C, S, D). **CK Computers** Plymouth 0752 780311. **Computer-fix** Camberley, Surrey 0276 66266. **Computer Repairs and Maintenance** Harrogate, Yorks 0423 65270. **Chiltern Computer Centre** Luton, Beds 0582 455684. **PC Engineering and Maintenance** 04867 88301. **Enfield Communications** Enfield, Middx 01 805 7772 (S, B, C, D). **Hemel Hempstead Computer Centre** Hemel Hempstead, Herts 0442 212 436 (S, B, C). **Micro Computer Trade Services** Southend 0702 62033/615809. **Micro-fix** London W10 01 968 9214 (B). **Micro-serv** Denny, Scotland 0324 823468. **Micro-surgery** Manchester 061 436 2688. **Micromend** Leeds 0532 792611. **MP Electronic Services** Dereham, Norfolk 0362 87327 (S). **Mancomp Ltd** Manchester 061 224 1888 (S). **Micro Repair Club** London 01 946 7777. **Micro Workshop** Muston, Manchester 061 205 4774 (S, C). **One Stop Micros** 230 Tottenham Court Road, London 01 631 5379. **Quantum Sinclair Service Centre** Coventry 0203 24632 (S, Z). **Suredata Computer Services** Edgware, Middx 01 951 0124. **Zedem Computers Ltd** Bedford 0234 213645 (S).

NET

QUESTIONS

While many micro computer magazines have regular comms sections, few set out to answer the multitude of problems that can arise when trying to establish a communications link between tow computers via a BT telephone line. From this month *E&CM* will carry a regular comms problem page that will set out to answer any technical problems that readers encounter – these can involve both micro to micro communication, problems associated with Prestel and other public databases, questions regarding the use of bulletin boards, or about the increasingly popular use of PSS (the Packet Switched System). Send any questions to the address shown on this page – we regret that we cannot provide individual answers to readers' queries, all replies will be via this column. To get the column going, this month we give answers to a selection of the letters we have received over the past few months.

Robert Schifreen is the man behind these comms pages. Send any news, gossip, information and questions to him at our editorial offices.



Q: I have a VTX 5000 modem on my Spectrum which I use to access Prestel and Micronet. Can you tell me what the Rx/Tx switch is used for?
A: Rx and Tx stand for Receive and Transmit respectively. The idea is that you can link your Spectrum and VTX 5000 to a friend's, and swap files down the telephone line without going via Prestel. The software to do this is now being given away free with new VTX modems. The data is sent down the line at 120 characters per second but, unlike Prestel, data can only go one way down the line at a time. So you and your friend have to decide who is going to send and who is going to receive before switching on the modems. To change direction, you both have to switch Rx/Tx modes. The Modem 1000, also from Prism, has the same features at the VTX, while the Modem 2000 can switch between receive mode and transmit mode under software control. Details on the Rx/Tx switch are scarce in the VTX manual, mainly because, in its early days Micronet didn't want subscribers to download software from the system and then to distribute copies to their friends using their modem!

Q: Can you tell me something about the Protek 1200 modem. It sounds really good value for money, but is it really?
A: It depends which micro you have. The Protek modem, launched last year, is an acoustic coupler complete with interface which allows access to Prestel on a Commodore 64, BBC model B or 48K Spectrum. At a cost of around £80 for the complete kit, it is good value for money, and the modem itself is certainly well made. But the software which comes with it came in for some criticism when it was announced. The Commodore and BBC version, while fairly basic in terms of facilities, are usable. The Spectrum version, in many people's opinion, is not. This is because the interface can't scan the telephone line for incoming characters and keep the screen updated simultaneously. A whole page is pulled down from Prestel at a time. Only when the entire page is in RAM is it then displayed on screen. This makes the system fairly slow and complicated to use, and takes up a lot of unnecessary on line time. Dynamic frames (Prestel's version of animation) are out of the question.

Q: Can a VTX 5000 modem be made to auto-answer?
A: Auto answer modems find their main use in Bulletin board systems. Left unattended, they will answer incoming calls, connect them to your micro and clear down the line when the caller hangs up. Your micro can then correspond with the caller's computer. There's no way that I know of making a VTX 5000 auto answer without extensive hardware modifications. Some Micronetters are currently forming a group dedicated to running a bulletin board based on a Spectrum. For more information, keep an eye on Chatline on page 811, their nightly meeting spot.

Q: Is it possible to get into non-Prestel systems, like MUD, using a Spectrum VTX 5000 modem.
A: The VTX modem has all its software built in, and this is custom designed for viewdata systems like Prestel. Data is received in "frames" which consist of 24 lines of 40 columns. When a frame is received, the screen is cleared before the next one is sent. Systems like MUD, which don't use Prestel standards, have normal scrolling displays, usually in 80-column format. To access MUD with a Spectrum, you have to use PSS (more details ab out this network system from BT on 01 920 0661) with a baud rate of 1200/75 and protocols of 7, even, 1. The text will be rather difficult to read, though, as it will start from the top left hand corner each time the bottom of the screen is reached.

However, for the measly sum of £2.95, you can download a program from Micronet which gives the VTX a full 40-column scrolling display compatible with MUD, PSS, Telecom Gold and more. Look in the telesoftware area for details – it's well worth it.

Q: I would like to get a modem for my BBC. Which is the best one to get?
A: This depends on what you want to do with the modem, and how much you want to pay for it.

A telephone line only has 2 wires. When you link your micro to a larger computer via a phone and a modem, the data has to travel down these 2 wires bit by bit, building up the characters at the other end. The speed at which these bits are sent is approximately equal to the baud rate. It doesn't matter what speeds you use, as long as both ends of the line are talking at the same speed. A character sent down a telephone line this way is made up, usually, of 10 bits. Therefore divide the baud rate by ten to determine the speed in characters per second.

Most bulletin boards in this country use a 300 baud system, while some commercial systems use 1200. Older teletypes go as slow as 110 or even 75 baud.

Prestel and Micronet use a 1200/75 baud rate. This means that data sent from Prestel to you comes at 1200 baud while characters which you send to Prestel (like keying in the number of a page) go at 75 baud.

If you want to be able to use all these different systems, you need a modem capable of running at various speeds. The Miracle Technology and Unicom models will both allow you to get into Prestel and bulletin boards, while the Acorn Prestel adaptor runs only at 1200/75 baud.

Prism's modem 1000 and Modem 2000 are mainly Prestel modems but, like the Spectrum VTX 5000, allow user-to-user file transfer at 1200 baud. The Modem 1000 has to be switched between receive and transmit manually, while the 2000 can do this under software control. A Micronet subscriber has even managed to use the 1200/1200 baud user to user feature of the modem 1000 as a bulk update terminal to Prestel (this is how the information providers actually update their pages – it would be far too slow sending all their information at 75 baud).

To use a modem with a Beeb also requires software. The Micronet ROM allows 1200/75 baud communication only, and is far from perfect as far as utilities go. Computer Concepts' Termi is good, as are Commstar and Communicator. These run at many baud rates, and have facilities to, for example, spool all incoming text to a disk file. The advantage of this is that some databases cost around £20 per hour to be online to. So, to read the day's news, spool it to disk at 1200 baud and print it out or read it later for free.

Incidentally, if you have a Prestel-type modem but aren't a subscriber to Prestel, dial up your local Prestel computer (get the number from Freephone Prestel Sales) and enter an ID of 4444444444 and a password of 4444.

NET

NEWS

Some bargains on modems, an X rated American bulletin board and a DIY Viewdata system – some of the highlights from this month's comms news.

● Following last month's sad demise of **Prism,** a company called **Modem House** kindly agreed to take most of Prism's unsold modems off their hands. Consequently, you should now be able to find the VTX 5000 modem for the Spectrum at under £50 – that's half price. The Modem 1000 and Modem 2000 are also around the £50 and £80 marks respectively. See advert in this issue.

● Talking of **VTX modems,** if you want to access "real" computers with a Spectrum, there is now a program to give a Spectrum with VTX modem a full 40-column scrolling display compatible with **Gold, PSS** and similar systems. It costs £2.95, and is downloadable from Micronet. Incidentally, the authors are working on a version which will allow you to spool incoming text to a Microdrive file. The greatest problem is – obviously – speed, but they're working on it.

● To see what a real comms newsletter should be like, you should join **American People/Link.** Available from Britain through PSS, the system is a **CompuServe** lookalike but at a fraction of the cost. Although it only started last Christmas, interest is already quite substantial. Being a member, I was sent their quarterly newsletter a couple of weeks ago, and, far from suffering a severe attack of blowing their own trumpet – like some other systems I could mention – People/link actually accept that they make mistakes. My favourite item was probably the announcement that, owing to a programming error, no customer was billed for connect time on a certain evening. "So", it says, "If you were online during that period please notify our accounting department. If the lines are busy, please keep trying". Access to People/Link is $2.95 an hour, billed, as are most systems in the States, direct to your credit card. More details from **MicroMouse** on **Micronet** page 800651.

● Again in America, **X-rated bulletin boards** are all the rage at the moment. Until now, they've been operated by individuals with weird senses of humour. But now, a company has started charging for them. An American magazine is charging $67.50 to sign up to **"Sextex",** with a 20-cents-a-minute online fee. The law forbids me from mentioning the subjects of their SIGs (special interest groups), but when I find out their PSS address I'll let you know.

● Again from the USA, where every call you make is itemised on your phone bill, I have heard of a subscriber whose phone bill for the last quarter was 173 pages long. It seems that someone managed to get hold of his credit card and used it to make 12,300 international calls over one weekend. And the total bill? Just $93,351.49. Wonder what the 49 cents were for?

● Back home again, the **CommunITel** are finally starting to deliver their Viewdata system after various software problems and non-working modems. The kit costs £375 and for your money you get a **DaCom modem** (list price £350), a disk of around 140K of software and all the manuals and leads that you need to run your own version of Prestel on a BBC micro. There's a good screen editor, some terminal software that allows you to access Prestel and, incidentally, to pinch frames from Prestel and use them in your database) and the remote software that lets callers access your system over the phone. There's also a **CET telesoftware formatter.** I was impressed with the whole setup, but unfortunately, not being in Victor Kiam's income bracket I couldn't quite afford to buy the company. But I do have a system up and running on which you can leave me messages. It's open from **8am to 11pm on 01 958 7098.** If you're a fan of **Microgrnome on Viewfax,** you may already know that Microgrnome also has a system. Known as the Gnome at Home, it's on **01 348 3247.**

● **Websters,** one of the largest micro software distributors in the country, has set up **MicroWeb.** They make extensive use of BT's electronic mail service, Telecom Gold, and Prestel. All the mobile sales team are equipped with hand-held micros, and each rep has a Telecom Gold mailbox. Daily bulletins can now be broadcast to the team instantly, and orders can be transmitted to the warehouse as soon as the rep can get to a phone. Retail outlets with Prestel facilities can access MicroWeb, Webster's pages on Prestel, to receive current news about what the company has in stock, and can also place orders on the system through response pages. Soon retailers will be able to download short trailers of current games from Prestel, to use as point-of-sale promotional material. Eventually the ordering service will be available to end users like you and I but, at the moment, it's just for retailers. If you use Prestel, take a peek at **page 2924115.**

● At the recent **LET** trade show, **ASCOM** were showing their new acoustic coupler for the Commodore 64. Complete with software, this is probably the most flimsy piece of equipment I've ever seen, and for £77.50 it's not cheap either. It's basically 2 flat cups held on to the handset with a piece of elastic. The modem only operates at 300 baud but more than that I don't know. The person manning the stand hadn't even heard of Prestel, and had no idea about how to use his own product. If you'd like to try getting something out of the company yourself, call **061 626 722.**

● **Electracom** have come up with one of those idea that you take one look at and say "Of course! – whjy didn't anyone think of it before?" It's a method for connecting micros in different parts of a building so that they can share data. Instead of using dedicated calbes or the telephone system, Electracom just use the normal ring main. The Micro's RS232 connectors are linked to special black boxes that do all the work, cutting out the need for any rewiring. As long as the equipment is mains powered and is all on the same phase, there's no problem. A "send" box and "receive" box is needed at each end, each of which costs £60. A switching box allows you to send data round different routes so that, for example, one printer can be shared by many micros. The select box costs £50. Details from 55 East St, Epsom or call **03727 43250.**

● Yet again, I've been promised that the **Unicom** (yawn) modem is now in full production. This week's figures quote a weekly production of 150 modems and, by the time that you read this, all the backlog will have been cleared.

● Incidentally, the **QL modem's** future is in dobut at the moment because **OEL** have called in the receiver. Let's hope that everything gets sorted out soon. With the QCALL auto-answer modem for the QL, and a 0.5Mb RAM pack, it should be possible to run a bulletin board on a QL without using disks.

A long running **legal battle in Los Angeles,** which has been on the lips of most Sysops for some time, finally ended recently. Thomas Tcimpidis, a 34 year old Sysop, was held responsible for the stolen credit card numbers which a caller had placed on his board. The case against Tcimpidis started in May '84 and was finally dropped through lack of evidence. Against the advice of his attorney, Tcimpidis is still operating his board.

If **Program Express** has its way, EDOS will be the word on everyone's lips this year. Not another new operating system for the QL, EDOS stands for **Electronic Distribution Of Software.** Retailers who fork out around £100 a week can have the centre of their shop obscured by a huge machine with tape and disk drive, connected by modem to a central computer in Edinburgh. If I wanted to buy a copy of a game for my micro, I would simply load a blank cassette or disk into the machine, pay the retailer the money, and a copy of the software would be transferred from the machine's internal 10Mb hard disk to my cassette or disk. The modem link is used to send up-to-the-minute software releases to all the stores. Also, the central computer can call the machines during the night and interrogate them, to see how much software has been sold. From this information it sends out a bill to the retailer every month. The idea is that the retailer is never overstocked or out of stock, each of which costs money. At the moment, around 80 software houses have agreed to have their wares sold in this way. With the dramatic reduction in overheads, there may even be some discounts with software bought like this.

● Since Program Express announced its intentions last year, **RoseTech** have launched **SoftStore.** Again, this allows the comsumer to load a blank cassette into the machine and get a copy of a game from a catalogue of software held on a hard disk. However, this system doesn't use modems. To update the system, a rep from RoseTech calls round with a disk pack under his arm! On the documentation front, RoseTech's machine prints out a small instruction sheet each time you buy a game. Programs like elite, with massive books of instructions, won't be available. Program Express send out separate stocks of documentation to the retailer which, as long as he doesn't run out of instruction books, should be all right.

● And that's about the end of our monthly miscellany of comms news for this time. If you have any interesting facts, or need advice, drop me a line to the magazine's editorial address, or send me an electronic message on **VISA.** If you make a habit of transatlantic conversations, I'm on CompuServe **(74106,1637** or **People Link (Hex Maniac).**

NET

This is a list of all known (legal!) databases available. If you know of any more, let us know. The list is split into sections, grouped under the baud rates at which they operate. Just look at the list which corresponds to the type of modem that you have, and get dialling!

NUMBERS

Note: Protocol is shown as number of bits per word, parity, number of stop bits. Bits per word can be 7 or 8, parity can be none (n), even (e) or odd (o) and stop bits can be 1 or 2. See your modem's instruction manual for details.

300 Baud Systems

Name: **BABBS**
Protocol: 8,n,1
Phone: 0225 23276
Times: Weekdays 9pm-midnight
Weekdays 9pm-9am
Notes: Atari based

Name: **BABBS 1**
Protocol: 8,n,1
Phone: 0394 276306
Times: 24hrs
Notes: Runs on an Apple but caters for everyone

Name: **BABBS 2**
Protocol: 8,n,1
Phone: 0269 778956
Times: 24hrs
Notes: Apple based

Name: **BASUG**
Protocol: 8,n,1
Phone: 0742 667983
Times: 24hrs day
Notes: British Apple Systems User Group

Name: **Bettisfield**
Protocol: 8,n,1
Phone: 094875 378
Times: 9am-9pm daily
Notes: Remote CP/M system

Name: **BITEC**
Protocol: 8,n,1
Phone: 0268 25122
Times: 24hrs

Name: **Birmingham North**
Protocol: 8,n,1
Phone: 0827 288810
Times: 24hrs
Notes: Runs on a Tandy. Strong MUD SIG

Name: **Bloxam**
Protocol: 8,n,1
Phone: 0295 720812
Times: Weekdays 7am-8am, 6pm-7pm
Weekends 2pm-6pm

Name: **CBBS-NE**
Protocol: 8,n,1
Phone: 0207 543555
Times: 2.30pm-9am daily
Notes: Mainly business users

Name: **CBBS Southwest**
Protocol: 8,n,1
Phone: 0392 53116
Times: 24hrs
Notes: Also runs at 1200/75 baud

Name: **CBBS Surrey**
Protocol: 8,n,1
Phone: 04862 25174
Times: 24hrs

Name: **Chiltern**
Protocol: 8,n,1
Phone: 07073 28723
Times: 9pm-8am daily

Name: **City BB**
Protocol: 8,n,1
Phone: 01 606 4194
Times: 24hrs
Notes: Also runs at 1200/75 baud

Name: **CNOL**
Protocol: 8,n,1
Phone: 0524 60399
Times: 24hrs
Notes: Mainly medical information

Name: **Communitree Hope**
Protocol: 8,n,1
Phone: 0874 711147
Times: 24hrs
Notes: Tree-structured. Very friendly. Caters for Welsh language.

Name: **Distel**
Protocol: 7,e,1
Phone: 01 679 1888 or 679 6183
Times: 24hrs
Notes: Run by Display Electronics. Remote ordering system.

Name: **Forum 80 Hull**
Protocol: 8,n,1
Phone: 0482 859161
Times: Tue/Thu 7pm-10pm
Weekends 1pm-10pm
Notes: The original BB, as run by Fred Brown.

Name: **Forum-80 Spa**
Protocol: 8,n,1
Phone: 0926 39871
Times: 1pm-midnight
Notes: Lots of info for Tandy users

Name: **Hamnet**
Protocol: 8,n,1
Phone: 0482 497150
Times: 24hrs

Name: **Livingstone BBS**
Protocol: 8,n,1
Phone: 0506 38526
Times: 24hrs

Name: **London Underground**
Protocol: 8,n,1
Phone: 01 863 0198
Times: 7pm-midnight daily

Name: **Mailbox-80 Warley**
Protocol: 8,n,1
Phone: 0384 635336
Times: Mon-Sat 5.30pm-8.30am
All day Sunday

Name: **Mailbox 80 Liverpool**
Protocol: 8,n,1
Phone: 051 428 8924
Times: 24hrs
Notes: Run by Peter "PCW" Toothill

Name: **Maptel**
Protocol: 8,n,1
Phone: 0702 552941
Times: 24hrs
Notes: Online ordering service for Maplin Electronics

Name: **MBBS**
Protocol: 8,n,1
Phone: 01 640 2617
Times: Thu-Sun 10am-10pm

Name: **Micro Live**
Protocol: 8,n,1
Phone: 01 579 2288
Times: 24hrs
Notes: Operated by the BBC Micro Live team.
Lots of useful notes.

Name: **Microweb**
Protocol: 8,n,1
Phone: 061 456 4157
Times: 24hrs

Name: **MOBBS**
Protocol: 8,n,1
Phone: 061 736 8449
Times: 24hrs
Notes: Also available at 1200/75 baud

Name: **OBBS**
Protocol: 8,n,1
Phone: 061 427 1596
Times: 24hrs

Name: **NBBS Chingford East**
Protocol: 8,n,1
Phone: 0692 630186
Times: Sun-Thu 10pm-7am
Fri-Sat 10pm-10am

Name: **PIP**
Protocol: 8,n,1
Phone: 0742 667983
Times: 24hrs
Notes: Operates on US Bell tones from midnight to 8am

Name: **Southern BB**
Protocol: 8,n,1
Phone: 0243 511077
Times: 24hrs

Name: **SITEC**
Protocol: 8,n,1
Phone: 0782 265078
Times: 24hrs
Notes: Stocke ITeC. Remote CP/M system.

Name: **TBBS London**
Protocol: 8,n,1
Phone: 01 348 9400
Times: 24hrs

Name: **TBBS Metro**
Protocol: 8,n,1
Phone: 01 341 7840
Times: 24hrs
Notes: Also runs at 1200/75 baud

Name: **TBBS Blandford**
Protocol: 8,n,1
Phone: 0258 54494
Times: 24hrs

Name: **WABBS**
Protocol: 8,n,1
Phone: 0903 42013
Times: 24hrs
Notes: Atari-based system

1200/75 baud systems

Note: Protocols for these systems is 7 bits, even parity, one stop bit unless otherwise stated. Format is either normal, scrolling screen display or Prestel-type viewdata display.

Name: **Aberdeen ITeC**
Format: Prestel
Phone: 0224 641585
Times: 24hrs
Notes: CommunITel viewdata system

Name: **Bildschirmtext**
Format: Prestel
Phone: 010 4930 1511
Times: 24hrs
Notes: Germany's Videotext system. Calls are EXPENSIVE!

Name: **Bulletin**
Format: Prestel
Phone: 0462 677177
Times: 24hrs
Notes: Public system run in ICL mainframe by Herts Council

Name: **CBBS Southwest**
Format: Scrolling
Phone: 0387 53116
Times: 24hrs
Notes: Public system run in ICL mainframe by Herts Council

Name: **CBBS Southwest**
Format: Scrolling
Phone: 0392 53116
Times: 24hrs
Notes: Also runs on 300 baud, 8,n,1

Name: **City BB**
Format: Scrolling
Phone: 01 606 4194
Times: 24hrs
Notes: Also on 300 baud, 8,n,1

Name: **C-View**
Format: Prestel
Phone: 0702 546373
Times: 24hrs
Notes: Run by Rochford County Council

Name: **Distel**
Format: Scrolling
Phone: 01 679 6183
Times: 24hrs
Notes: Also on 300 baud, 8,n,1

NEW SERIES
THE FIRST SPECTRUM WORDPROCESSOR IN FIRMWARE
PART TWO

OF MICE & JOYSTICKS

Richard Sargent describes how the hardware of the *E&CM* integrated wordprocessor may be connected to mice, trackballs or joysticks.

In order to provide a full featured word processor for the Spectrum, including the option of driving the WP from a joystick or trackball, it is necessary to construct a small interface circuit. The main components of this interface are shown in **Table 1.**

The unusual chip in this set is the ADC0844 and it is this which allows the use of trackball, mouse or analogue joystick. This facility is, strictly speaking, an optional extra. The software will check to see if the IC if fitted, and if it isn't, well, it's back to cursor control using the arrow-keys. In fact the saving in terms of cost is considerable. At the time of writing the ADC is £6.95, the joystick £3.25 and the trackball £18. However, there are side benefits associated with the ADC and more will be said about this later on.

The other ICs are all essential and amount to some £10 in total. The PIO handles the centronics printer port, reads four auxiliary switches, and turns the EPROM on and off whenever necessary. The memory decoder maps the EPROM at addresses 2000H to 3FFFH (but only when the PIO permits it) and so the EPROM blanks off the last 8K of the Spectrum's resident operating system. As to the new EPROM, the choice has to b e the 8K device: it is easier to decode than the 4K chip and may soon be cheaper as well. Such is progress.

Overlay ROM

Two ROMs can never co-reside in the same address space without disrupting each other's control over the CPU. However, the ROMCS pin on the Spectrum edge connector allows the internal ROM to be effectively disconnected at any time. Normally the signal which enables the new ROM is the same signal which switches off the internal ROM and the CPU is not aware that a change-over has taken place. There is a note of caution to be sounded here. The WP hardware will switch out the Spectrum ROM correctly, but there is no guarantee that it will switch out any other ROMs which might be plugged into the back of your Spectrum! The advice is, if you have a Disk system, Microdrives or Wafadrives, done't connect the WP until you've found out exactly wheat provision (if any) has been made by the manufacturers for *their* shadow ROM to be switched out.

Most commercial overlay systems monitor the code flowing through address 0008H, which is the start of the BASIC error-trapping routine. They look for the asterisk character which signals that the shadow ROM should be switched in to deal with the new command, of which the asterisk is part. The shadow ROM may be anywhere in the 0 to 3FFF area.

The WP deals with the shadow EPROM in a much more straightforward way. A small BASIC utility routine turns the WP ROM on, and it stays on for the duration except on three occasions – reading files, writing files, and printing to the ZX printer. Fortunately there's nothing much in the now dormant Sinclair ROM from 2000H onwards to interest a wordprocessor, except the character set (!), and even that can be relocated in RAM.

Now the ADC

The A/D facility needs to be considered at this early stage: there are about four types of case which would be suitable for the WP

TABLE 1

A 2764 EPROM
A Z80A PIO
A 74LS138 Memory decoder
An ADC0844 A/D converter

electronics and which you buy or build will depend very much on whether you decide that the analogue port is worth having.

The ADC0844 converts four channels of analogue data to a resolution of 8 bits. It includes a few variations in the way in which the analogue inputs can be grouped together, and alll configurations can be used, from machine code or BASIC, with or without the presence of the WP EPROM. Three modes of operation are permitted. They are differential, single ended and pseudo-differential. It is also possible to configure two channels to differential mode, and two channels to single-ended mode! This makes the chip sound complicated – in fact nothing is further from the truth. The ADC can be thought of as a black box having four inputs, each of which will accept a voltage between 0V and +5V. Each voltage is presented to the computer as a number between 0 and 255, and in the WP application the input is a simple voltage taken from the wiper of a joystick's potentiometer. The internal working of the chip is, as the manufacturer's functional diagram **(Figure 1)** shows, highly complex, but the external connections are very straightforward. Three control lines, RD, WR and CS communicate with the X80 CPU, the first two directly, and CS via a logic circuit which ANDs the CPU's TODO

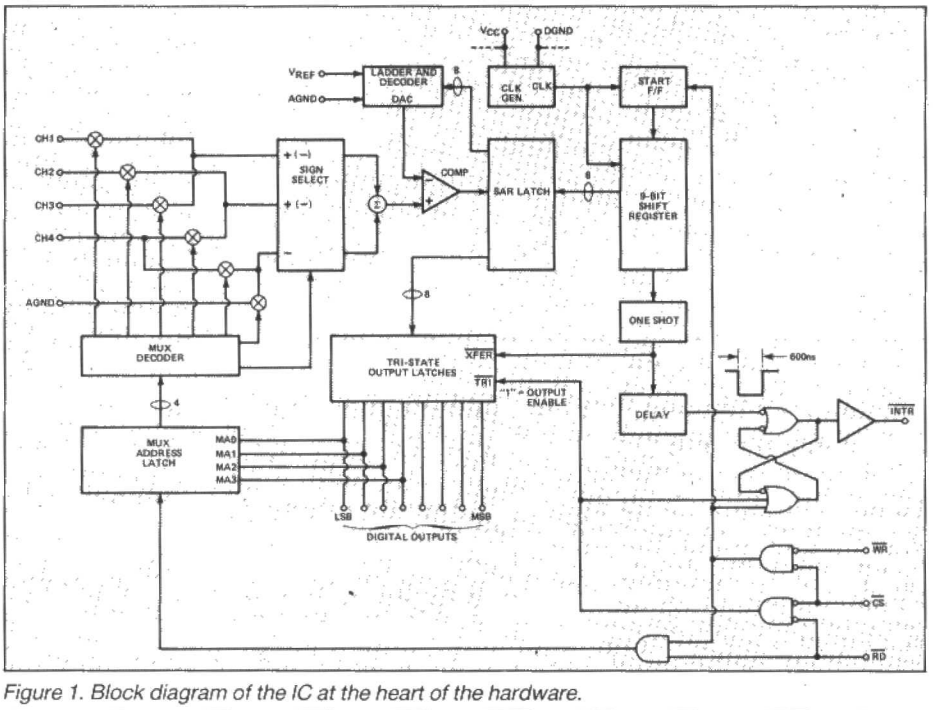


Figure 1. Block diagram of the IC at the heart of the hardware.

and all signals together to create the chip select signal. Eight data lines link directly to the CPU's Data Bus, and four analogue inputs look at the world outside. The Vref pin may be tied to +5V and the analogue ground AGND may be tied to logic ground, 0V. The chip doesn't require a clock signal and negative voltage supplies are not needed either. A WRITE to the ADC selects the channel you wish to read, and the READ grabs the data: it's as simple as that. The chip will indicate on its INTR pin when it has finished converting the voltage to a number (which it will do in 40uS), but if you're using BASIC the conversion will be ready and waiting by the time the interpreter gets around to asking for it.

The ADC is decoded on the Spectrum's I/O address line A6 which gives it an address of 65471. To select Channel 0

OUT 65471,4

and then to read it

PRINT IN 65471

Channelks 1, 2 and 3 are selected by codes 5, 6 and 7. Codes 0, 1, 2, 3, C, D and E are used to select differential and pseudo-differential modes of operation.

Trackballs & Joysticks

These two-potentiometer devices come in a variety of shapes and sizes. A joystick can be made for the price of two 100k potentiometers and some considerable ingenuity and manual dexterity on the part of the constructor. This is the Heath Robinson option. Maplin sell a 2-axis joystick which requires some plastic or metal-cutting skills if it's to be cased neatly. It *can* be fixed straight onto the Spectrum case using double-sided sticky-pads! Going upmarket slightly brings into consideration the commercial devices widely available for the BBC microcomputer, but which can be used for this project. If you buy a joystick, make sure it really is an analogue type – just because it fits into the BBC analogue port doesn't mean that it is necessarily an analogue joystick! Acorn and others market the real thing. If you buy one of these, the wordprocessor electronics can be housed in a square box, and the joystick plugged into it using standard "D" series connectors. Trackballs are also not always what they seem. The expensive sort have opto-electrical insides and will not run on an analogue port. There is at least one, however, which fits the bill. Magenta sell a BBC Trackball Controller which has the necessary potentiometers (5k types), and which has a case of its own large enough to take the wordprocessor PCB.

If you can borrow a joystick or trackball then you can start to evaluate it just by using the ADC0844 chip and a small amount of BASIC. **Listing 1** shows a short program which puts Xs all over the Spectrum screen. **Listing 2,** in Z80 source code, shows how all the ADC channels can be read by just a few bytes of machine code. But what of the two unused channels? They can be used to drive a 16-key auxiliary keypad for your Spectrum, and that can be most useful, especially in wordprocessor applications.

The full circuit details, including the PCB pattern for Spectrum WP will be given next month.

LISTING 1

```
10 LET oldc=128: LET oldr=128
12 OUT 65471,4
14 LET r=IN 65471
16 OUT 65471,5
18 LET c=IN 65471
20 LET row=(oldr+r)/2
21 LET oldr=r
22 LET col=(oldc+c)/2
23 LET oldc=c
24 LET col=255-col
26 LET row=255-row
28 IF row>210 THEN LET row=210
30 PRINT AT 0,0;row;"   "
32 PRINT AT 1,0;col;"   "
34 PRINT AT row/10,col/10;"x"
36 GO TO 12
```

LISTING 2

```
0017                ;
0017                ;
0017                ;READ ALL 4 ANALOGUE LINES
0017                ;
0000                      ORG 0
8000                RAM   EQU 8000H
8000
0000 01BF08         SET_UP LD BC,08BFH  ;Reg B holds counter
0000                                    ;Reg C holds port address
0003 211700               LD HL,CODES
0006 110080               LD DE,RAM
0009 EDA3           DO_IT OUTI          ;Start conversion by
0009                                    ;issuing correct code
000B EB                   EX DE,HL
000C 3E0F                 LD A,15       ;Software delay value
000E 3D             WAIT  DEC A
000F 20FD                 JR NZ WAIT
0011 EDA2                 INI           ;Read and store result
0013 EB                   EX DE,HL
0014 20F3                 JR NZ DO_IT
0016 C9                   RET;urn to calling routine
0016
0017 04050607 CODES       DB 4,5,6,7
```

REVEALING THE BBC MICROS

BUILT-IN MODEM

Paul Beverley shows that, with the right software, the BBC micro's cassette and RS423 ports may be merged to provide the basic functions of a modem.

One of the problems with computer communications is that some communications channels such as the public telephone system cannot accept data directly in the form of the RS423 voltage levels. Audio tones have to be used instead to represent the data. This normally requires the use of a modem (MODulator-DEModulator) which translates between the voltage levels that the computer uses and the audio tones that the telephone system can transmit. The trouble is that modems tend to be a bit expensive, starting at round about the £50 mark.

So it might surprise you to learn that the BBC micro (along with most home computers) already has a built-in modem: at least it has a system which translates the digital data in the computer into audio tones and vice versa – it's the cassette interface! If we can harness the hardware of the cassette interface we should then be able to send data on any communications system that can transmit audio tones. For example, the radio hams amongst you could, for a few pence, link in to your transmitters and start talking to BBC microcomputer owners around the country or across the world.

The ability to harness the cassette interface for this purpose depends on a very clever software technique described in the Advanced User Guide (Bray, Dickens & Holmes). In this article it is simplified, but follows their basic idea, which we acknowledge with gratitude. The technique involves the use of *FX calls to link together the RS423 serial interface and the cassette interface. This is possible because, as I explained in the first article in this series, both interfaces use the 6850 ACIA, the input and output lines of which are switched by the serial ULA chip which itself is software controllable.

Once this link is made in software, any data sent by your program to the RS423 port will, instead of appearing as changing levels on the RS423 connector, appear as audible tones on the cassette output. If these tones are received by another computer on its cassette input then, if the same software link has been made, it will transfer the characters which it has received across into the RS423 input buffer. So we can use any of the character input and output commands, such as GET, INKEY and PRINT etc, and can transmit and receive characters using the method described last month for a direct RS423 link.

The software

The same software can be used in a number of different hardware situations. When the program is first run it asks whether the user wants to start by sending or receiving. This seems a sensible way to arrange things since you would normally establish the link verbally and then connect up the computers, then you can decide who will transmit first.

The program has the facility for sending pre-prepared text by using the *EXEC command, and if you want to keep a copy of the received text you can use *SPOOL. However this is only possible on disk-based computers, because the computer is already using its cassette hardware to establish the transmission link and cannot therefore use it as a file system. But since you are not actually using the cassette file system software, the disk file system remains active and can read and write data from the disk within the program. The spooling is controlled by the sending computer; the sequence would be *Press shift/f0*. This causes the receiving computer to do a "*SPOOL RECTEXT" which opens a file called "RECTEXT". *Press shift/f2*. This does a "*EXEC MESSAGE" ie it takes the file called "MESSAGE" and sends it down to the other computer. *Press shift/f1*. This causes the receiving computer to do a "*SPOOL" which closes the file.

Program analysis

The program is in many ways similar to the two we looked at last month, but with the addition of special commands to control the cassette interface and the 6850 ACIA. The initialisation procedure (lines 650 to 730), apart from asking whether to send or receive, sets up the link between the RS423 and the cassette (720).

The sending procedure (160 – 430) starts by switching the cassette motor relay off. This is only needed when the communication system requires a switch to receive and transmit, for example an intercom or a radio transmitter. The two *FX138,1,1 commands (180, 190) put two dummy characters into the RS423 input buffer so that by changing the size of the buffer in software (lines 240 and 420) the cassette pilot tone can be switched on and off.

The *FX2,2 (line 200) selects the keyboard as input while enabling the RS423 input buffer. The sending speed is then selected at line 210. The next two commands (lines 220, 230) have the effect of resetting the 6850 ACIA and selecting the correct clock speed to give 300 baud transmission on the cassette interface. The cassette tone is then switched on (240) and a "[" character and a space are sent to the screen only (250, 260). The RS423 output is enabled (270) and then the keyboard input buffer flushed (280). This deletes any characters accidentally typed in while you were waiting patiently for the other person to stop chatting and let you have a go!

Within the REPEAT loop, the computer picks up characters from the keyboard, displays them on the screen and sends them to the receiving computer. If shift and function key 2 are pressed then the ASCII character produced is interpretted as a

"... Radio hams among you could start talking to BBC micro owners around the country or world-wide! ..."

LISTING 1.

```
 10 REM RS423 2-way Communication
 20 REM via the Cassette Interface
 30 REM with disc file transfer.
 40 REM (C) 1984
 50 REM Norwich Computer Services
 60
 70 MODE 3
 80 PROCinitialise
 90 IF receive PROCreceive
100 REPEAT
110   PROCsend
120   PROCreceive
130   UNTIL0
140 END
150
160 DEF PROCsend
170 *MO.0
180 *FX138,1,1
190 *FX138,1,1
200 *FX2,2
210 *FX8,3
220 *FX156,3,252
230 *FX156,2,252
240 *FX203,9
250 *FX3,0
260 PRINT'"[ ";
270 *FX3,5
280 *FX21,0
290 REPEAT
300   D%=GET
310   IF D%=130 D%=0: *EXEC MESSAGE
320   IF D%=93 VDU32
330   IF D%>0 VDU D%
340   IF D%=13 VDU10
350   UNTIL D%=93
360 REPEAT
370   UNTIL ADVAL(-3)>&BE
380 TIME=0
390 REPEAT
400   UNTIL TIME=50
410 *FX3,0
420 *FX203,255
430 ENDPROC
440
450 DEF PROCreceive
460 *MO.1
470 PRINT'"[ ";
480 *FX7,3
490 *FX156,3,252
500 *FX156,2,252
510 *FX21,1
520 *FX2,1
530 REPEAT
540   D%=GET
550   IF D%=93 D%=125
560   IF D%=91 D%=123
570   IF D%<128 VDU D%
580   IF D%=128 THEN *SPOOL RECTEXT
590   IF D%=129 THEN *SPOOL
600   UNTIL D%=125
610 VDU7
620 *FX2,2
630 ENDPROC
640
650 DEF PROCinitialise
660 VDU19;4;0;
670 PRINT"Send or Receive?"
680 S=GET
690 receive=0
700 IF S=82 OR S=114 receive=1
710 ON ERROR PROCerror:END
720 *FX205,64
730 ENDPROC
740
750 DEF PROCerror
760 *FX205,0
770 *FX2,0
780 *FX3,0
790 REPORT:PRINT" at line ";ERL
800 ENDPROC
```

signal to select a file called "MESSAGE" from the disk and send it donw to the receiving computer.

The 93 code ("]") is detected (lines 320 and 350) because it is used to signal the end of the transmission. At this point the program comes out of that loop and goes into the loop (at lines 360 and 370) which waits until the RS423 output buffer has been emptied. Then an extra half second delay is generated (380 – 400) before switching the RS423 output off (410) and switching off the pilot tone (420).

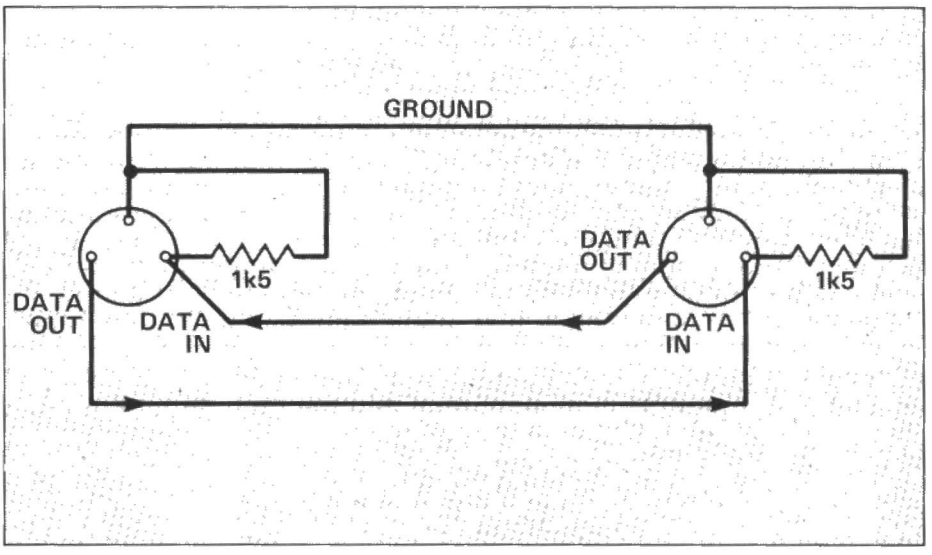
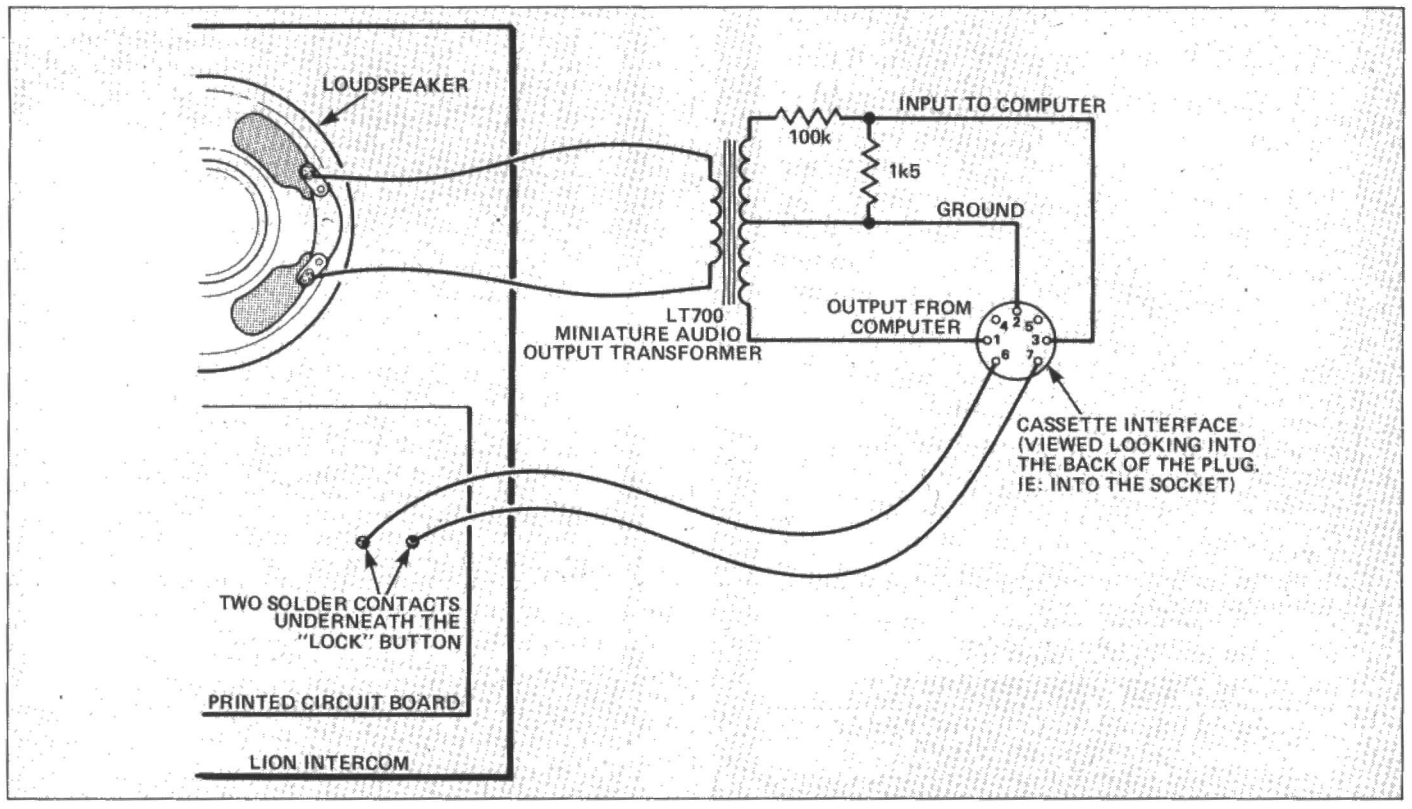


Figure 1 (right). Back to back cassette lead connections.

Figure 2 (below). Connections to a LION FM mains intercom.



The receive procedure (450 – 630) prints a curly bracket on the screen to indicate the start of the received text. The receive speed is selected (480) before the ACIA is reset (490, 500). Then the RS423 input buffer is flushed to remove the dummy characters that were inserted into it during the sending procedure. The RS423 is then selected as the input channel (520), and the receiving loop entered (530 – 600). If a square bracket (open or closed) is received it is changed into a curly bracket to indicate that it is received text, and only characters less than 128 are actually printed out (570). If a 128 is received, this initiates the transfer of all text to a disk file named "RECTEXT" (580), and a 129 closes this spool-file. The end of the loop comes with a closed curly bracket, and the user is given a bleep (610) to say that at last it is his turn to speak! The keyboard input is then re-selected (620).

may feel that this is bad practice, but it is done because when you are in the receive mode the keyboard is disabled anyway and the only two keys recognised are escape and break.

Extending the software

If you want to have short messages ready to send, you can use the function keys. For example:

```
721  *KEY 0 Hello there!:M
722  *KEY 1 This is Fred speaking.:M
723  *KEY 2 Please could you
724  *KEY 3 Thank you for
etc.
```

For a printed copy of the data being transferred then switch the printer on at the start of PROCreceive by using VDU 2 and switch it off again at the beginning of the PROCsend with VDU 3. The printer would of course have to be one which uses the parallel printer port since the serial interface is being used for the communication link. An alternative would be to use function keys 3 and 4 with the shift key pressed. These generate ASCII codes 131 and 132, and so in the send routine you could add:

```
302  IF D%=131 VDU2
304  IF D%=132 VDU3
```

Then you would have to change line 540 in the receive routine so that it did not act on these as received codes, as follows:

The disk drive can be controlled for spooling text when in the send mode, ie by adding:

```
306  IF D%=128 D%=0:*SPOOL
     RECTEXT
308  IF D%=129 D%=0:*SPOOL
```

The hardware

The simplest way to test the software is to make up a back-to-back cassette lead and have the two computers next to each other. You only need to use 3-pin (180 degree) DIN plugs since the other pins are not needed – 3 pin plugs are easier to solder than the 5 or 7-pin plugs. As you can see from **Figure 1** the output from one plug has to be connected across to the input of the other and vice versa, whilst the earth connection just goes straight across. In addition to these connections a resistor must be inserted between the earth line and the data output at each end. If you use a physically small resistor and are fairly neat and careful it can be mounted inside the plug itself. The resistor adjusts the relative phase of the two tones which are used to represent the two logic levels and makes it easier for the receiving computer to discriminate between them. (Now you know!).

"... The trick is to link the cassette and RS423 interfaces with *FX calls ..."

The error routine (750 – 800) simply resets the RS423 and cassette systems to normal and prints out an error message. No method has actually been provided to break out of this program other than pressing the escape key which brings you out through this routine. Some programmers

VIDEO PRINTER

In part two of John Yau's project he describes the software that controls the picture grabbing process and a Mode 0 viewer for assessing picture quality before printing.

The heart of the video digitiser system lies within the software which sees the video picture right from the start of the signal chain in the video camera or recorder to the final hard-copy poster produced by the dot matrix printer. Two poster sizes are available; 0.9m x 0.56m and 1.8m x 1.12m approximately. Restriction in printer paper width means that the poster has to be printed in vertical segments and assembled. The poster sizes are chosen to preserve the correct aspect ratio of the original picture to within 1%.

Four programs are presented to support the digitiser hardware. The first is the frame grab program, which captures the video picture onto disk. In order to be able to view the stored picture, either for errors or its suitability for printing, a Mode 0 viewer program has been written. Using this program, the picture file can be viewed on the BBC Micro partially at full captured resolution or wholely at reduced resolution. Next in the process is the printer driver, which reads the picture file and prints out the poster segments using a bit image mode. To support the printer driver the fourth program generates 128 grey scale characters into a random access file.

Both the Mode 0 Viewer and printer driver programs have facilities to alter the grey scale character look-up table. Hence there is scope for image processing experimentation, such as observing the effects of reduced bit resolution and image enhancement.

The frame grab program

This program shown in **Listing 1** captures a 192 x 200 picture in 7-bit resolution. Such a picture would require 38400 bytes which would seem to present a problem since the BBC Micro's memory capacity is much less than this! The problem can be overcome by digitising the left half of the picture into a 19200 byte buffer in the BBC Micro, saving it onto disk and then digitising the second half in the same manner. For reasons which will be explained later the dimensions and size of the picture file is of special importance in the Mode 0 viewer and printer driver programs.

The frame grab process commences by pushing the reset button SW1 to set the delay data to 255. If only 6 bit resolution is required then it is possible to do the latter by software if the spare port bit is configured as a reset line.

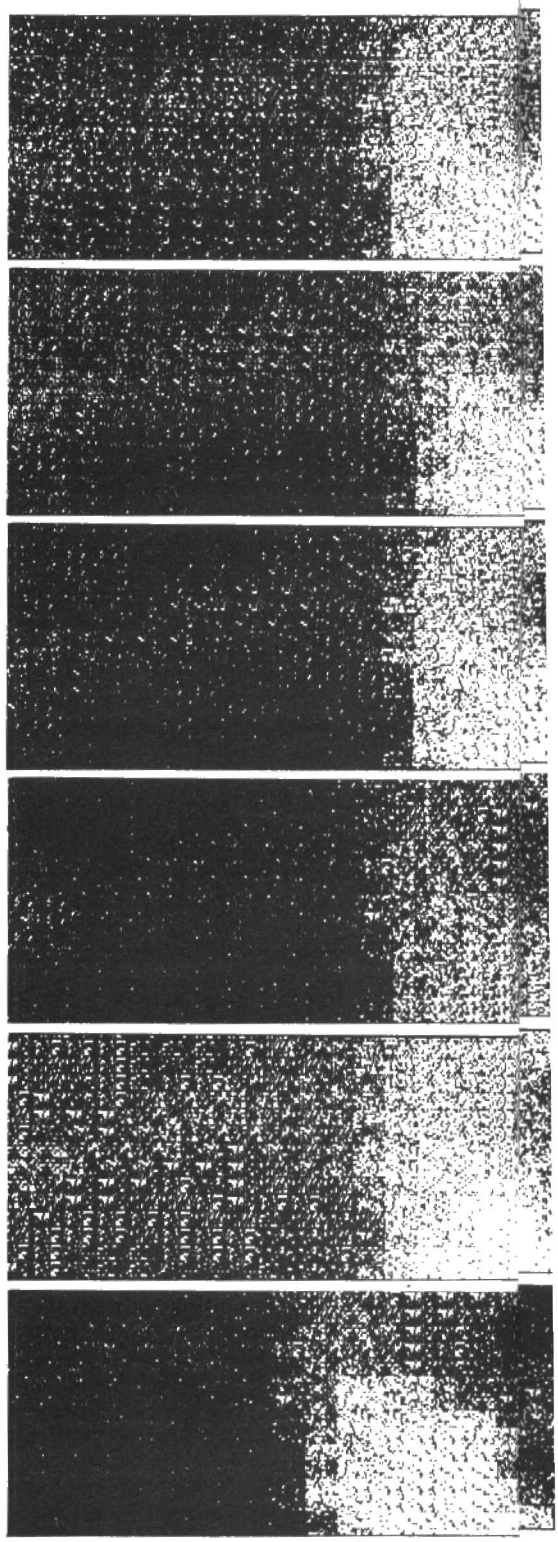
When a frame sync pulse is detected, by examining bit 0 of the user port, the BBC's user VIA is programmed so that the detected line sync pulses can interrupt via the CB1 line. The interrupt routine then continues collecting the latched data from the ZN448 convertor until a complete picture column has been placed in the buffer. Following this, the VIA is programmed so that CB1 is no longer allowed to interrupt. The program then decrements the delay counter by pulsing CB2 once, locks onto the next frame sync pulse (módulo two)

"... The heart of the system lies in software ..."

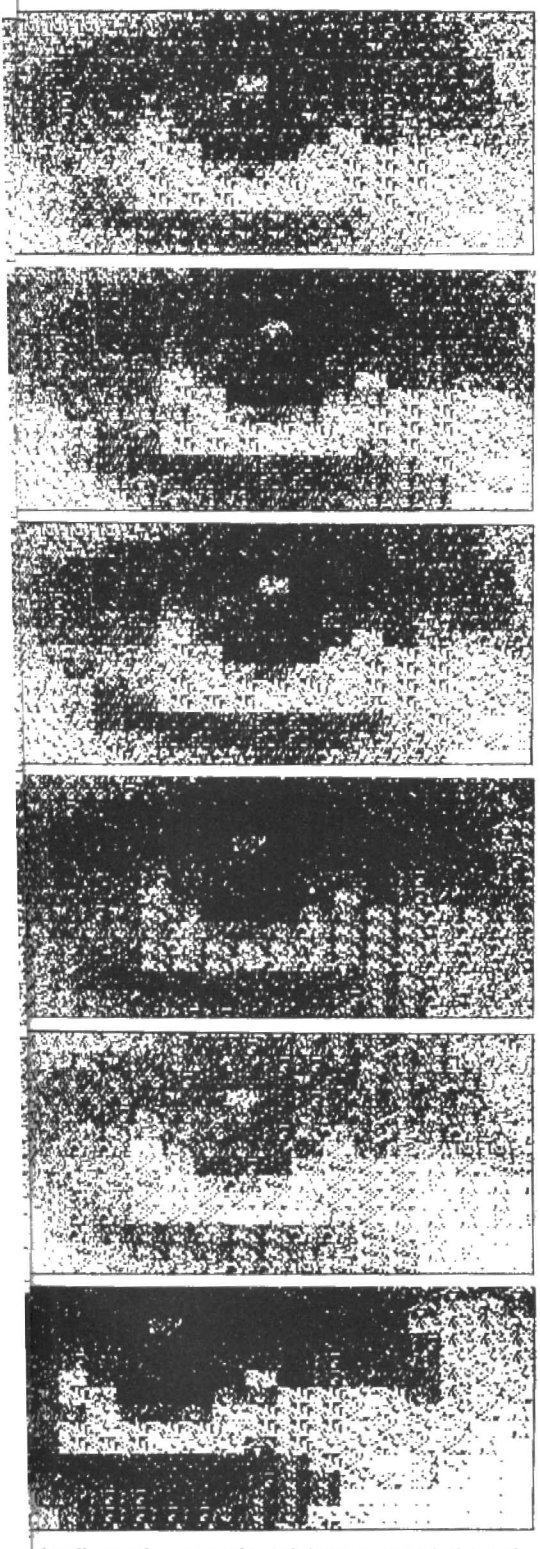
and continues the process until the buffer is full. When this happens it is emptied by saving onto disk and the second half of the picture is then digitised.

It is important to note that the data collection interrupt routine has the highest priority because, during its active period, the BBC's system interrupts are shut down. This is necessary to avoid chaos caused by the BBC's background timer interrupts contesting with the user routine.

The final result is two 19200 byte files, comprising of the first and second half of the captured picture, having the specified filename with the extenders '1' and '2' for the respective files. In order for the Mode 0 viewer program to work, two files have to follow each other as far as sectors on the disc are concerned, so it is best to start off with a clear disk to ensure this. An important point to note is that whilst the grab program locks onto every second field frame during the capture of a picture half, it may be possible that the disk drive access time for saving the first half causes the second half to lock onto the wrong field frame. There was no such problem with the prototype but if that is a positional difference of one line between the picture halves



Six print outs of the same video frame that illustrate the step parameters within the printer driver routine.



the effects of varying the brightness, threshold and

&3000	&3008	&3278
&3001	&3009	
&3002	&300A	
&3003	&300B	
&3004		
&3005		
&3006		
&3007		
&3280		
&3281		
&7B06		
&7B07		
&7D80		
&7D81		
&7D82		
&7D83		
&7D84		
&7D85		
&7D86		
&7D87	&7D8F	&7FFF

Figure 1. Screen memory map of the BBC computer's graphics mode.

the solution is to keep adding a small Basic delay after saving the first picture-half until the problem is rectified.

The Mode 0 Screen Viewer

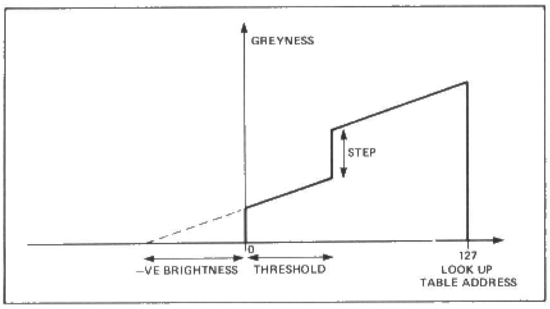
This program, **Listing 2** uses Mode 0, the highest resolution graphic mode of the BBC Micro (640 x 320) to display the captured picture files. The latter are read from disk sector by sector, and for each byte of pciture data, an 8 x 2 pixel grey scale character is plotted at the appropriate position on the screen. At full resolution only the selected upper or lower half of the screen is displayed. If the whole picture is displayed every second column and every second row is ignored, so the picture is only viewed at 1/4 resolution. In order to view the full length of the picture and not be restricted by the graphic mode selected, the screen 'wraps' round itself. Although its use is intended for checking the picture file en route from the grab program to the printer driver program, it can be used to obtain direct screen dump files of pictures. In fact the viewer program along with the grab program is in itself a self contained software package for the video digitiser.

Since the program reads sectors and does not distinguish between files the two halves of a picture must lie adjacent to each other on the disk. Also the start sector of the picture file must be known (*INFO will reveal this). The program reads a sector at a time into a 256 byte buffer. A machine code routine is then called which processes the contents of this buffer and plots the appropriate grey characters on the screen. This continues until the picture is plotted.

The machine code routine calculates the next memory mapped screen location and plots a character if required. Its complexity is due to the fact that the screen memory map arrangement of the graphics mode is rather unsuited for our application (see **Figure 1**). An offset has to be added at every eighth location in order to address the next screen row. Also the location of the top of a column, if temporarily stored, has to be retrieved and incremented by eight when advancing to the next column. When a picture is viewed, the sectors of the picture file are continuously read, but the column does not advance until the specified start column has been reached.

What are the reasons for selecting a picture file of 192 x 200, a total size of 38400 bytes? Primarily this is because 19200, the half picture file size, is an integral multiple of 256, the number of bytes in a sector. This means that the two halves of the picture lie on one contiguous mass of sectors, ie it avoids any gap at the end of the last sector of the first picture half. More practically, 19200 is about the largest buffer one can get on the BBC Micro's mode 7, allowing for memory for the residing grab program.

A simple software contrast control is implemented by adding an offset to the data byte before it references the grey character table. When a sector is read, the machine code routine deals with each byte in turn. A data byte is first limited and then the four MSB bits are extracted from it by means of rotation operations. The contrast offset is then subtracted from the data. Finally the data is limited again (eg any data



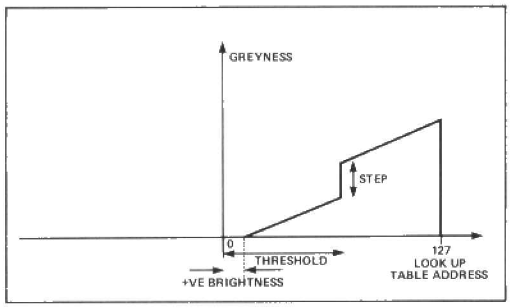


Figure 2. The effect of the software contrast controls. Brightness moves the graph left or right thus controlling the overall amount of black and white. Step and threshold can be used to 'silhouette' a picture. More on this next month.

LISTING 1: Frame grab pattern

```
10 REM * Video Frame Capture Program
20 REM * (C) John Yau December 1984
30 REM * stores picture as 192 X 200 7bit elements
40 MODE7
50 DIM C% 250,CBLOCK% 18,NAME% 7,VBUFF% 19200
60 REM *
70 REM * zero page parameters
80 REM *
90 TEMP=&70
100 YSTART=&71
110 YEND=&72
120 YSIZE=&73
130 DCNT=&74
140 NCOLS=&75
150 ROWCNT=&76
160 COLCNT=&77
170 SYSTVEC=&78
180 STORE=&7A
190 REM *
200 REM * user VIA locations
210 REM *
220 ADCPRT=&FE60
230 DCYPRT=&FE61
240 DDRB=&FE62
250 DDRA=&FE63
260 ACR=&FE6B
270 PCR=&FE6C
280 IROFR=&FE6D
290 IRQER=&FE6E
300 REM *
310 REM * system VIA IER location
320 REM * & file save OS call
330 SYSIER=&FE4E
340 OSFILE=&FFDD
350 REM *
360 REM * assemble machine code
370 REM *
380 FORpass%=0TO2 STEP2
390 P%=C%
400 [OPT pass%
410 \*
420 \* user VIA initialisation
430 \*
440 LDA #0
450 STA DDRB \ set ADCPRT input port ( ORB )
460 LDA #255
470 STA DDRA \ set DCYPRT output port ( ORA )
480 LDA #2 \ set ADCPRT to latch upon active transition of CB1
490 STA ACR
500 LDA #16 \ CB1 low to high transition sets CB1 interrupt flag
510 STA PCR
520 \*
530 \* steal system IRQ vector and suppress
540 \* system interrupts whilst frame grab
550 \* in operation
560 \*
570 JSR IRQSTEAL
580 \*
590 \* main program for frame grab
600 \*
610 CLC
620 LDA YSIZE \ calculate and store limits of window
630 ADC YSTART
640 STA YEND
650 LDA #0
660 STA COLCNT \ initialise column counter
670 JSR VADDR
680 LDX DCNT
690 BEQ NXTCOL
700 .DLY
710 JSR DELDEC
720 DEX
730 BNE DLY
740 .NXTCOL
750 LDA #0
760 STA ROWCNT \ zero row counter
770 JSR DELDEL
780 LDX #2 \ get every second frame
790 .SYNC
800 LDA ADCPRT \ test for frame sync
810 AND #1 \ reset on bit 0
820 BNE SYNC \ no sync
830 .WAIT1
840 LDA ADCPRT \ wait for frame sync positive going edge
850 AND #1
860 BEQ WAIT1
870 DEX:BNE SYNC
880 LDA #144 \ enable ADC to interrupt via CB1
890 STA IRQER
900 .WAIT2
910 LDA ROWCNT \ wait for completion of column conversion,when ROWCNT=YEND
920 CMP YEND
930 BNE WAIT2
940 LDA #16
950 STA IRQER \ dis-able CB1 interrupts
960 INC COLCNT \ increment for next column
970 LDA COLCNT
980 CMP NCOLS \ all columns completed?
990 BNE NXTCOL
1000 \*
1010 \* video frame grabbed
1020 \*
1030 \* restore system interrupts
1040 \*
1050 JSR IRQREPLACE
1060 RTS
1070 .DELDEC
1080 LDA #192+16:STA PCR
1090 LDA #224+16:STA PCR
1100 RTS
1110 \*
1120 \* Interrupt service routine
1130 \*
1140 .IROT
1150 PHA:TXA:PHA
1160 LDX ADCPRT \ clear IRQ flag
1170 LDA ROWCNT \ if start row not reached yet then
1180 CMP YSTART \ increment row count and exit
1190 BCC CONT1
1200 CMP YEND \ if end row is reached the do nothing
1210 BEQ EXIT
1220 TXA \ read ADC value
1230 LDX #0
1240 STA (STORE,X) \ store in video buffer
1250 LDA #1 \ calculate address for next element
1260 CLC
1270 ADC STORE
1280 STA STORE
1290 BCC CONT1
1300 INC STORE+1
1310 .CONT1
1320 INC ROWCNT \ increment for next row
1330 .EXIT
1340 PLA:TAX:PLA
1350 RTI
1360 :
1370 .IRQSTEAL
1380 SEI \ dis-able interrupts
1390 LDA &FE4E \ save system VIA IER status
1400 STA TEMP
1410 LDA #127 \ turn off system IRQs
1420 STA &FE4E
1430 LDA &0204 \ save system vector
1440 STA SYSTVEC
1450 LDA &0205
1460 STA SYSTVEC+1
1470 LDA #IROT MOD 256 \ direct IRQV1 to user IRQ routine
1480 STA &0204
1490 LDA #IROT DIV 256
1500 STA &0205
1510 CLI \ enable interrupts
1520 RTS
1530 :
1540 .IRQREPLACE
1550 SEI
1560 LDA TEMP \ restore IER status
1570 STA &FE4E
1580 LDA SYSTVEC \ restore system vector
1590 STA &0204
1600 LDA SYSTVEC+1
1610 STA &0205
1620 CLI
1630 RTS
1640 :
1650 \ put start address of video buffer in STORE
1660 .VADDR
1670 LDA #VBUFF% MOD 256
1680 STA STORE
1690 LDA #VBUFF% DIV 256
1700 STA STORE+1
1710 RTS
1720 :
1730 .VSAVE \ block memory save
1740 LDA #0 \ OSFILE code for block save
1750 LDX #CBLOCK% MOD 256
1760 LDY #CBLOCK% DIV 256
1770 JSR OSFILE
1780 RTS
1790 ]
1800 NEXTpass%
1810 REM *
1820 REM * end of m/c assembly
1830 REM *
1840 PROCFD("Video Digitiser Program")
1850 PRINT "Input filename ";:INPUT F$
1860 IF LEN(F$)>7 THEN 1850
1870 FORI=0TO1:FORJ=0TO6:PRINTCHR$(1[illegible]);CHR$157;:NEXT:PRINT:NEXT
1880 PRINT"Press 'G' to start grab. "
1890 REPEAT:A$=GET$:UNTIL A$="G"
1900 PRINT"Grabbing frame....."
1910 PROCcapture(F$)
1920 PRINT"Frame captured.
1930 END
1940 :
1950 DEFPROCcapture(N$)
1960 ?DCNT=20:?NCOLS=96
1970 ?YSTART=[illegible]:?YSIZE=200
1980 !CBLOCK%=NAME%
1990 CALL C%
2000 $NAME%=N$+"1"
2010 !(CBLOCK%+10)=VBUFF%
2020 !(CBLOCK%+14)=VBUFF%+?NCOLS*?YSIZE
2030 CALL VSAVE
2040 ?DCNT=[illegible]
2050 CALL C%
2060 $NAME%=N$+"2"
2070 !(CBLOCK%+10)=VBUFF%
2080 !(CBLOCK%+14)=VBUFF%+?NCOLS*?YSIZE
2090 CALL VSAVE
2100 ENDPROC
2110 DEFPROCFD(A$):FOR I%=1 TO 2:VDU 141,&9D84;&BD83;:PRINTSPC
     (16-LENA$DIV2)A$:NEXT:
ENDPROC
```

LISTING 2. Mode 0 screen viewer

```
  10 REM * Mode 0 Video Screen
  20 REM * Capture Viewer
  30 REM * resolution 80x100, 16 shades
  40 REM * (C) John Yau December 1984
  50 REM *
  60 DIM L% 250,CHAR1% 16,CHAR2% 16,BUFF% 256,PAR% 10,NM% 20
  70 !(PAR%+1)=BUFF%:PAR%?5=3
  80 PAR%?6=&55:PAR%?9=&21
  90 MCNT=&70:RCNT=&71:CCNT=&72:CPLT=&73:CTRAST=&74:SKIPC=&75
 100 STCOL=&76:QUIT=&77:RESFLG=&78:QUIT=&79:STORE=&7A:STRTCOL=&7C:NCOLS=&8F
 110 PROCassemble
 120 PROCloadcharset
 130 *FX15,0,230
 140 MX=-1:BUFFER=&00:LOWER=&FF:F$="PICFILE":?CTRAST=8:SEG=LOWER
 150 *KEY255
 160 MODE7:PROCcls
 170 PRINT"Please select:"
 180 VDU31,0,3:PRINT"Contrast setting (range 0-15) is
 190 VDU31,0,4:PRINT"Screen save filename :";F$
 200 PRINT"M    - Return to Mode 7 menu."
 210 PRINT"U    - Increase contrast."
 220 PRINT"D     Decrease contrast."
 230 PRINT"I     *INFO Disc"
 240 PRINT"V    - View picture file."
 250 PRINT"Q    - Quit loading picture."
 260 PRINT"F     Specify screen save filename."
 270 PRINT"      (default is 'PICFILE')"
 280 PRINT"S     Save screen onto disc"
 290 PRINT"      (active only on Mode 0)"
 300 REPEAT
 310   VDU31,0,25:PRINT"";
 320   REPEAT:A$=GET$:UNTILAr  ""
 330   IFA$=" "*"THEN370
 340   VDU31,1,22
 350   *FX13B,0,42
 360   INPUT S$:PROCsystemcall(S$):GOTO160
 370   IFA$="D"THEN 400
 380   ?CTRAST=?CTRAST+1:IF?CTRAST>15 ?CTRAST=15
 390   VDU31,35,3:PRINT15-?CTRAST
 400   IFA$<>"U"THEN 440
 410   IF?CTRAST=0 THEN 440
 420   ?CTRAST=?CTRAST-1
 430   VDU31,35,3:PRINT15-?CTRAST
 440   IFA$<>"V"THEN480
 450   PROCcls
 460   PRINT"Which drive ";:INPUT DRV:IF DRV>3 THEN 460
 470   PRINT"Type in start sector ";:INPUT O$
 480   IFO$=""THEN470
 490   SCSTRT=EVAL(O$)
 500   PRINT"Start column (pic size 192x200) ";:INPUT ST
 510   IF(ST<1)OR(ST>110)THEN500
 520   ?STCOL=ST:?NCOLS=192-ST
 530   PRINT"Whole or Partial picture ";
 540   REPEAT:A$=GET$:UNTIL(A$="W")OR(A$="P")
 550   IFA$="W" THEN570
 560   ?RESFLG=0:?STCOL=?STCOL DIV 2:GOTO610
 570   ?RESFLG=1
 580   PRINT:PRINT"Upper or Lower picture? (U/L)";
 590   REPEAT:A$=GET$:UNTIL(A$="U")OR(A$="L")
 600   IFA$="L" SEG=LOWER ELSE SEG=UPPER
 610   MODE0
 620   PROCloadpic
 630   REPEAT:A$=GET$:UNTILA$<>""
 640   IFA$="Q"THEN VDU19,1,ASC(A$)-230;0;:GOTO630
 650   IFA$="S" PROCsavescreen
 660   IFA$="M" THEN160
 670   GOTO630
 680   IFA$="I" PROCsystemcall("*INFO F.*":GOTO160
 690   IFA$<>"F"THEN730
 700   PROCcls
 710   PRINT"Input new screensave filename';:INPUTF$
 720   GOTO160
 730   UNTIL FALSE
 740 END
 750 :
 760 DEFPROCloadpic
 770 ?MCNT=0:?RCNT=0:?CCNT=0:?CPLT=0:?SKIPC=SEG:?QUIT=&00
 780 !(LOADE+3)=&3400:?STRTCOL=ST:?STORE=ST-8
 790 REPEAT
 800   PROCLoadsector(SC,DRV):SC=SC+1
 810   A=!BUFF%(0):IFA$="Q" ?QUIT=&FF
 820   VDU31,0,0:PRINT"COLUMN NO.=";
 830   D.?&70,?&71,?&72:PRINT?CCNT ELSE PRINT2*?CCNT
 840   UNTIL(SC=LAYOUT+?QUIT+&FF)
 850 UNTILFALSE
 860 :
 870 DEFPROCloadsector(SC,DR)
 880 PAR%=DR:PAR%?1=SC DIV 10:PAR%?8=SC MOD 10
 890 A%=&7F:X%=PAR% MOD 256:Y%=PAR% DIV 256:CALL &FFF1
 900 ENDPROC
 910 :
 920 DEFPROCassemble
 930 FORpass%=0TO2 STEP2
 940   P%=L%
 950   OPTpass%
 960   LDX #0
 970   .LOOP
 980   LDA BUFF%,X \ get pixel from sector buffer.
 990   CMP #126 \ limit max value to 127.
1000   BCC NHIGH
1010   LDA #126
1020   .NHIGH
1030   AND #124
1040   CLC:ROR A:ROR A \ convert to 4 bit data.
1050   SEC:SBC CTRAST \ subtract contrast offset.
1060   BCS OTHRES
1070   LDA #0 \ limit min to 0.
1080   .OTHRES
1090   CMP #15 :
1100   BCC UTHRES
1110   LDA #15 \ limit max to 15.
1120   .UTHRES
1130   TAY:TXA:PHA \ pixel data is now the Y index into grey table.
1140   LDA CHAR1%,Y \ buffer index also saved.
1150   LDX #0
1160   STA (STORE,X) \ plot top half of grey scale char.
1170   LDA CHAR2%,Y
1180   LDY #1
1190   STA (STORE),Y \ plot bottom half of grey scale char.
1200   PLA:TAX \ restore buffer index.
1210   INC MCNT \ every 8th screen address needs to have &072A added to
1220   LDA MCNT \ to point to next picture element of column.
1230   CMP #4
1240   BEQ NXTBLK \ branch if modulo 8th screen address.
1250   CLC:LDA STORE \ if not modulo 8 just add 2 tp screen address.
1260   ADC #2:STA STORE
1270   BCC CONT2
1280   INC STORE+1
1290   .CONT2 CLC:BCC CONT1 \ unconditional branch.
1300   .LINE BNE LOOP
1310   .NXTBLK CLC
1320   LDA #0:STA MCNT \ reset modulo 4 counter ( counts characters
1330   LDA #&7A:ADC STORE:STA STORE \ add &072A to screen address.
1340   LDA #&0C:ADC STORE+1:STA STORE+1
1350   .CONT1
1360   INC RCNT \ increment row count.
1370   LDA RCNT
1380   CMP #100 \ 100th pixel plotted yet? ( column complete )
1390   BNE EXIT
1400   LDA #0:STA RCNT:STA MCNT \ yes, so reset counters.
1410   LDA #&10:SBC STORE
1420   STA STORE \ for every modulo 2 column do not advance to next
1430   BCC SKIPLP
1440   INC CCNT:LDY CCNT \ increment column counter.
1450   CPY STCOL:BCC NSTRT \ if start column not reached yet do not
1460   INC CPLT \ screen address to next column.
1470   LDA CPLT
1480   CMP NCOLS \ if last column plotted, set quit flag to exit.
1490   BNE NFIN
1500   LDA #&FF:STA QUIT
1510   .NFIN
1520   LDA #0 \ load to advance column.
1530   BNE SKIPLP \ unconditional branch.
1540   .NSTRT
1550   LDA #0 \ load for non advance of column.
1560   .SKIPLP
1570   CLC:ADC STRTCOL
1580   STA STRTCOL:STA STORE
1590   LDA #0:ADC STRTCOL+1
1600   STA STRTCOL+1:STA STORE+1
1610   .EXIT
1620   LDA STORE:CMP #(&2A00+&278) MOD 256
1630   BNE PCESS
1640   LDA STORE+1:CMP #(&2A00+&278) DIV 256
1650   BNE PCESS \ reset screen address if screen is full
1660   LDA #&00:STA STRTCOL:STA STORE
1670   LDA #&2A:STA STRTCOL+1:STA STORE+1
1680   .PCESS
1690   LDA RESFLG \ if 1/4 resolution whole picture is selected then
1700   BNF NLPS \ increment X twice so that only every second piece
1710   INX \ of data is read.
1720   .NLPS
1730   INX
1740   BNE LINE \ if sector buffer not all read then continue.
1750   RTS
1760   ]
1770   NEXTpass%
1780 ENDPROC
1790 :
1800 DEFPROCsavescreen
1810 X%=NM% MOD 256:Y%=NM% DIV 256
1820 $NM%="*SA. "+F$+" 3000 8000"
1830 CALL &FFF7
1840 ENDPROC
1850 :
1860 DEFPROCsystemcall(O$)
1870 VDU31,0,23
1880 X%=NM% MOD 256:Y%=NM% DIV 256
1890 $NM%=O$:CALL &FFF7
1900 PRINT"Press any key.":REPEAT:A$=GET$:UNTILA$<>""
1910 ENDPROC
1920 :
1930 DEFPROCPD(A$):FORI%=0TO1:VDU&9D84;&8083;:PRINTSPC(16-LENA$DIV2)A$
1940   DEFPROCcls:CLS:PROCPD("Mode 0 Video Screen Viewer"):ENDPROC
1950   :
1960   DEFPROCloadcharset
1970   FORI=0TO15:READ CHAR1%?I,CHAR2%?I:NEXT
1980   DATA 255,255,255,191,223,251,237,223,107,191,217,175,218,35
1990   DATA 90,165,37,216,90,33,38,80,148,64,18,32,32,4,0,64,0,0
2000   ENDPROC
```

going lower than zero stays zero). The resultant processed byte is then used as an index into the grey character look-up table. The character is plotted and the next screen address is calculated. The process is repeated until the buffer is empty and the next sector is ready to be read.

If a direct screen save is desired the required filename should be specified first (by selecting menu choice). The picture is then viewed and pressing 'Q' for 'quit' aborts any further sector reading. This prevents the screen wrapping round. Note that once the picture has loaded the red function keys can be pressed to change the colour of the screen (blue is best for RGB monitors whilst white is most effective for UHF). Pressing 'S' will then save the screen onto disk under the specified filename. To load such a saved picture, simply enter code 0 and *LOAD the file.

This program is essential if picture posters are to be taken from a video recorder. The quality of the composite video signal from a VCR is not exactly broadcast standard and varies from machine to machine. Noisy or distorted frames are not unknown, hence the viewer program ensures that a picture is a good 'take' before commitment to the printing process.

NEXT MONTH

Full listings of the printer driver and grey scale look up table. The author of this article will be providing copies of the software on disk – ordering information next month.

68 COMPUTING CORNER

This month sees the addition of a new regular feature to the pages of *E&CM*. 68 Computing will concentrate on features, software and projects aimed at users of the 68xx family of processors. This spread of editorial coverage encompasses computer systems from the humble 8-bit Dragon computer to the icon laden Macintosh taking in Sinclair's QL enroute. We'll also be paying close attention to the progress of the 16-bit Atari ST machines and to Commodore's activity in this area. A recent 'Byte' editorial said that 1985 could be the year in which the 68xxx MPUs finally emerge from the shadow of the Intel/Zilog processors that have dominated both the 8- and 16-bit micro computer industry to date. *E&CM* shares this view, this will be the year of the 68000 and 68 computing will be here to reflect the growing interest in this group of processors.

The 68xxx family has enjoyed a considerable reputation for technical superiority when compared to its rivals; the same cannot be said when it comes to measuring the performance of the processors with their commercial success. The reasons for this owe little to any technical factors, rather than rival manufacturer's products have had the knak of being in the right place at the right time. In the case of the 6809 vs both the 6502 and the Z80, while the '09 is regarded by many machine code programmers as vastly superior, it came to the market a little late and was not available when the first micro computers were at the design stagew. This bith the original PET computer and the Tandy TRS80 ignored the 6809 in favour of the 6502 and the Z80 respectively. This sealed the fate of the 6809 in a world where being first is everything. Subsequent generations of micro computer adopted one or other of the established processors consigning the 6809 to the status of an also ran in terms of commercial success.

The same fate loooked likely to befall the 68000. When IBM adopted the Intel 8088 and derivatives in favour of the Motorola MPU it looked as if history was to repeat itself. Fortunately the 68000 managed to find some important supporters before it was too late. The Apple Macintosh and the Sinclair QL are the most important 68K machines to date but the Atari ST machines and the Commodore Amiga system are following the 68000 route. This it is the 68000 rather than the 8088 that looks set to dominate the low end of the 16-bit computer market.

The 68xxx series of processors, because of their attractive performance in the eyes of software engineers, have long been associated with elegant operating systems. The 6809 supports both OS9 and FLEX, two powerful operating systems that are the envy of anyone brought up with CP/M. In the case of the 68000 it is UNIX that offers programmers a powerful software environment. Digital Research's GEM (Graphics Environment Manager) is another very exciting tool soon to be available to users of the 68000 MPU. To this impressive list of operating systems must be added QDOS, Sinclair's own enironment for the QL computer.

From the above it will be seen that 68K computing is likely to be one of the most exciting areas of computing over the next few years. 68 Computing will be here to report on all developments in this field.

OS9 STARTUP

Michael Graham with hints on customising your OS9 system disk.

Listing the files on the OS9 system disk will show, in addition to the OS9Boot, commands and system directories a file named 'startup'. Startup is an OS9 command file and is automatically run when the system is booted. Listing the file as supplied on the system disk will produce the output shown in **Listing 1.**

The listing shows that the startup command file is simply a text file containing a sequence of OS9 commands. By modifying the file it is possible to customise the system disk to suit your own specific requirements.

Some possible modifications to the file include the addition of the GO51 command, to put the computer's display into the software generated 51 column mode, and bypassing the SETIME command. The Real Time Clock provided by OS9 suffers from the fact that it often displays anything but the real time. This is because the clock is driven from a 50Hz interrupt within the computer. If any activity that dissables the interrupts is initiated, for example reading or writing to a disk file, the clock is effectively stopped. This makes the clock of somewhat questionable use and the need to enter the time at system startup is sometimes annoying. Although simply pressing the return key in response to the enter time prompt will suffice it is possible to remove even this requirement.

Simply it would seem that it is only necessary to remove the SETIME command line from the startup file. As usual this solution throws up more problems than are evident at first sight. This is because the SETIME command must be issued when the system is boosted if the multitasking capabilities of OS9 are to perform correctly.

Instead of deleting the SETIME command therefore some other solution must be found. The powerful indirection operators of OS9 provide the answer. The SETIME command as it stands expects data from the keyboard by virtue of the fact that it is followed by the /<term operator. Changing this to </D0/time will instead cause the command to use a file named time on drive 0 for its input. This file can either consist of a string of six zeros, a space and a further six zeros or, preferably, a 'real' date eg 850101 as the first string and then six zeros – this because simply entering zeros for the date will cause the date stamping feature in some application software, eg Dynacalc, to trown up annoying question marks.

The startup file can most easily be altered by using the Stylograph word processor but it can equally well be modified using the OS9 command BUILD. The time file is created in a similar fashion.

The amended startup file is shown in **Listing 2.** Booting the system up with this version of startup installed will put the system straight into the 51 column mode without the need for the user to input any time/date information.

The above demonstrates just how easy it is to modify the operation of the OS9 operating system to suit the requirements of individual users.

LISTING 1

```
printerr/d0/sys/errmsg
setime </term
```

LISTING 2

```
printerr/d0/sys/errmsg
setime </d0/time
go51
```

QL UTILITY

FUNCTION KEY PROGRAMMER

Adam Denning provides the means of programming the QL's function keys with general purpose strings.

The five function keys on the left of the QL seem to be fairly useless, don't they? Although they each return a distinct character code and can be used in combination with CTRL and SHIFT to get even more codes, there's no way of programming them to reproduce long, complicated strings a la BBC Micro.

Oh yes there is!

The key to programming function keys is the 50Hz interrupt which scans the keyboard and updates the clock. Whenever the keyboard is canned and a key is being pressed, it is inserted into a keyboard buffer. The buffer chosen by QDOS is the current keyboard queue, which is at an arbitrary address pointed to by the system variable SV_KEYQ. This holds the address of the queue attached to the currently active keyboard channel, which is generally channel 0 but could be anything if other jobs are running.

All we need to do is to supply our own 50Hz interrupt routine which looks at the last character entered into the current keyboard queue. If this character is a function key (or any other key, for that matter), then we must insert our desired string into the buffer. To keep things in order and to keep other programs happy, we should leave the function key code in the buffer rather than remove it. We will soon find out just how we can link another 50Hz interrupt routine into the system as a QDOS extension.

Of course, we'll also want to be able to program these keys, and the most convenient place from which to program them is the SuperBasic interpreter. This means that we have to write a procedure which accepts strings for a key and assigns them to the designated key.

The keyboard queue pointed to by SV_KEYQ is a straightforward serial I/O queue, used as a circular buffer by the system. The first sixteen bytes consist of four pointers, of which at least three are immediately useful. These pointers are shown in **Table 1** (below).

Keyboard buffer + 0:	address of next queue
Keyboard buffer + 4:	address of end of queue
Keyboard buffer + 8:	address of next position to write to
Keyboard buffer + 12:	address of next position to read from
Keyboard buffer + 16:	the buffer itself

As the buffer is circular, we need to take a bit of care when we're examining it, to make sure that we don't fall off the end. The address pointed to by the end of the queue is not used for storage.

There is a QDOS vectored utility routine called IO_QIN which inserts the byte held in D1 into the queue pointed to by A2. If the queue is full, which means that the character cannot be entered, the routine returns with ERR_NC.

"... The QL function keys appear useless ... not any more ..."

To link a 50Hz interrupt routine into the list of interrupt routines, we take advantage of a QDOS manager trap. The particular one we're going to use is MT_LPOLL. It needs only one piece of information passed to it – the address of the link. The link is a two long-word piece of memory, the first four bytes of which hold the address of the

LISTING 1

```
LOC      OBJECT    STMT   SOURCE STATEMENT

                     1 * A routine to allow the QL function keys to be used as 'soft' keys
                     2 * By Adam Denning (C) 1985 Adam Denning
                     3 * Started 11th March 1985
                     4
=0002804C            5 SV_KEYQ  EQU   $2804C      Address of current keyboard queue
                     6
                     7 * Offsets into keyboard queue
                     8
=0000                9 NEXTQ    EQU   0          Address of next keyboard queue
=0004               10 ENDOFQ   EQU   4          Address of the end of the queue
=0008               11 WRITEQ   EQU   8          Address of next position to write to
=000C               12 READQ    EQU   12         Address of next position to read from
                    13
=001C               14 MT_LPOLL EQU   $1C        QDOS routine to link polled task
=00E0               15 IO_QIN   EQU   $E0        QDOS routine to put byte in a queue
=0110               16 BP_INIT  EQU   $110       QDOS routine to install Basic procedures
=0116               17 CA_GTSTR EQU   $116       QDOS routine to fetch args as strings
=FFF1               18 ERR_BP   EQU   -15        QDOS 'bad parameter' error
                    19
0000' 43FA 001A     20 START    LEA.L   PROC_DEF,A1
0004' 3478 0110     21          MOVE.W  BP_INIT,A2
0008' 4E92          22          JSR     (A2)
000A' 41FA 001E     23          LEA.L   KEY_LINK,A0
000E' 43FA 0022     24          LEA.L   KEY_INT,A1
0012' 2149 0004     25          MOVE.L  A1,4(A0)
0016' 701C          26          MOVEQ   #MT_LPOLL,D0
0018' 4E41          27          TRAP    #1
001A' 4E75          28          RTS
                    29
001C' 0001          30 PROC_DEF DC.W    1
                    31
001E' 008A          32          DC.W    KEYS_PROC-*
0020' 03            33          DC.B    3
0021' 4B45 59       34          DC.B    'KEY'
                    35
0024' 0000 0000 0000 36         DC.W    0,0,0
                    37
002A' =000B         38 KEY_LINK DS.L    2
                    39
0032' 2479 0002 804C 40 KEY_INT MOVE.L  SV_KEYQ,A2
0038' 226A 0008     41          MOVE.L  WRITEQ(A2),A1
003C' 204A          42          MOVE.L  A2,A0
003E' D1FC 0000 0010 43         ADDA.L  #16,A0
0044' B3C8          44          CMPA.L  A0,A1
0046' 6604          45          BNE.S   KEYCONT
0048' 226A 0004     46          MOVE.L  ENDOFQ(A2),A1
004C' 5389          47 KEYCONT  SUBQ.L  #1,A1
004E' 47FA 00C2     48          LEA.L   KEYTAB,A3
0052' 41FA 00D2     49          LEA.L   KEYDEFS,A0
0056' 1211          50          MOVE.B  (A1),D1
0058' 6726          51          BEQ.S   ENDREPL     Avoid an odd feature!
005A' 0C01 00E2     52          CMPI.B  #$E2,D1     CTRL-CAPS LOCK ?
005E' 660E          53          BNE.S   KEYSRON
0060' 12BC 00E1     54          MOVE.B  #$E1,(A1)
0064' 7401          55          MOVEQ   #1,D2
```

LISTING 1 (Continued)

```
0066' 41FA 00A8          56          LEA.L   FUNCSON,A0
006A' B550               57          EOR.W   D2,(A0)
006C' 4E75               58          RTS
006E' 43F8 0040          59 KEYSHON  LEA.L   FUNCSON,A1
0072' 4A51               60          TST.W   (A1)
0074' 6608               61          BNE.S   ENDREPL
0076' 9213               62 KEYLOOP  CMP.B   (A3),D1
0078' 6700               63          BEQ.S   KEYCHANGE
007A' 508B               64          ADDQ.L  #4,A3
007C' B1C9               65          CMPA.L  A3,A0
007E' 66F6               66          BNE.S   KEYLOOP
0080' 4E75               67 ENDREPL  RTS
                         68
0082' 7600               69 KEYCHANGE MOVEQ   #0,D3
0084' 3628 0002          70          MOVE.W  2(A3),D3        Offset of key string
0088' 49FA 009C          71          LEA.L   KEYDEFS,A4
008C' D9C3               72          ADDA.L  D3,A4           A3 now holds true address
008E' 7800               73          MOVEQ   #0,D4
0090' 381C               74          MOVE.W  (A4)+,D4        D2 holds string length
0092' 5344               75          SUBQ.W  #1,D4
0094' 68EA               76          BHI.S   ENDREPL
0096' 121C               77 SWAPKEY  MOVE.B  (A4)+,D1
0098' 3078 00E0          78          MOVE.W  IO_QIN,A0
009C' 4E90               79          JSR     (A0)
009E' 4A80               80          TST.L   D0
00A0' 66DE               81          BNE.S   ENDREPL
00A2' 51CC FFF2          82          DBRA    D4,SWAPKEY
00A6' 60DA               83          BRA.S   ENDREPL
                         84
                         85 * A SuperBasic extension to program the function keys
                         86
00A8' 3478 011A          87 KEYS_PROC MOVE.W  CA_GTSTR,A2
00AC' 4E92               88          JSR     (A2)
00AE' 665E               89          BNE.S   EXIT_KEYS
00B0' 70F1               90          MOVEQ   #ERR_BP,D0
00B2' 0C43 0002          91          CMPI.W  #2,D3
00B6' 6656               92          BNE.S   EXIT_KEYS
00B8' 7200               93          MOVEQ   #0,D1
00BA' 7400               94          MOVEQ   #0,D2
00BC' 3436 9800          95          MOVE.W  0(A6,A1.L),D2
00C0' 1236 9802          96          MOVE.B  2(A6,A1.L),D1
00C4' 0401 0030          97          SUBI.B  #'0',D1
00C8' 0C01 0001          98          CMPI.B  #1,D1
00CC' 6D40               99          BLT.S   EXIT_KEYS
00CE' 0C01 0005         100          CMPI.B  #5,D1
00D2' 6E3A              101          BGT.S   EXIT_KEYS
00D4' 5301              102          SUBQ.B  #1,D1
00D6' 5242              103          ADDQ.W  #1,D2
00D8' 0242 FFFE         104          ANDI.W  #-2,D2
00DC' D3C2              105          ADDA.L  D2,A1
00DE' 3436 9802         106          MOVE.W  2(A6,A1.L),D2
00E2' 0C42 0050         107          CMPI.W  #80,D2
00E6' 6E26              108          BGT.S   EXIT_KEYS
00E8' 363C 0052         109          MOVE.W  #82,D3
00EC' C2C3              110          MULU    D3,D1
00EE' 41FA 0036         111          LEA.L   KEYDEFS,A0
00F2' D1C1              112          ADDA.L  D1,A0
00F4' 30C2              113          MOVE.W  D2,(A0)+
00F6' 4A42              114          TST.W   D2
00F8' 6712              115          BEQ.S   EMPTYKEY
00FA' 5242              116          ADDQ.W  #1,D2
00FC' 0242 FFFE         117          ANDI.W  #-2,D2
0100' 5342              118          SUBQ.W  #1,D2
0102' 30F6 9804         119 MOVELOOP MOVE.W  4(A6,A1.L),(A0)+
0106' 5489              120          ADDQ.L  #2,A1
0108' 51CA FFF8         121          DBRA    D2,MOVELOOP
010C' 7000              122 EMPTYKEY MOVEQ   #0,D0
010E' 4E75              123 EXIT_KEYS RTS
                        124
0110' 0000              125 FUNCSON  DC.W    0
                        126
                        127 * A table of key codes
                        128
0112' E800              129 KEYTAB   DC.B    $E8,0          key F1
0114' 0000              130          DC.W    KEYSTR1-KEYDEFS
0116' EC00              131          DC.B    $EC,0          key F2
0118' 0052              132          DC.W    KEYSTR2-KEYDEFS
011A' F000              133          DC.B    $F0,0          key F3
011C' 00A4              134          DC.W    KEYSTR3-KEYDEFS
011E' F400              135          DC.B    $F4,0          key F4
0120' 00F6              136          DC.W    KEYSTR4-KEYDEFS
0122' F800              137          DC.B    $F8,0          key F5
0124' 0148              138          DC.W    KEYSTR5-KEYDEFS
                        139
                        140 * A table of key definitions
                        141
0126'                   142 KEYDEFS
0126' =0052             143 KEYSTR1  DS.W    41
0178' =0052             144 KEYSTR2  DS.W    41
01CA' =0052             145 KEYSTR3  DS.W    41
021C' =0052             146 KEYSTR4  DS.W    41
026E' =0052             147 KEYSTR5  DS.W    41
                        148
                        149          END
```

next routine in the list (this is filled in by QDOS), and the next four bytes hold the address of the routine itself. Once the link has been established, our routine will get called every 50th of a second, just like the keyboard scan.

Before we can write the entire program, we need to decide upon a few things. Let's say that we want a maximum of 80 characters per function key definition, and that we'll have this space permanently allocated in the resident procedure area lalong with the interrupt routine and the additional SuperBasic procedure. This procedure is called KEY, and is used as follows:

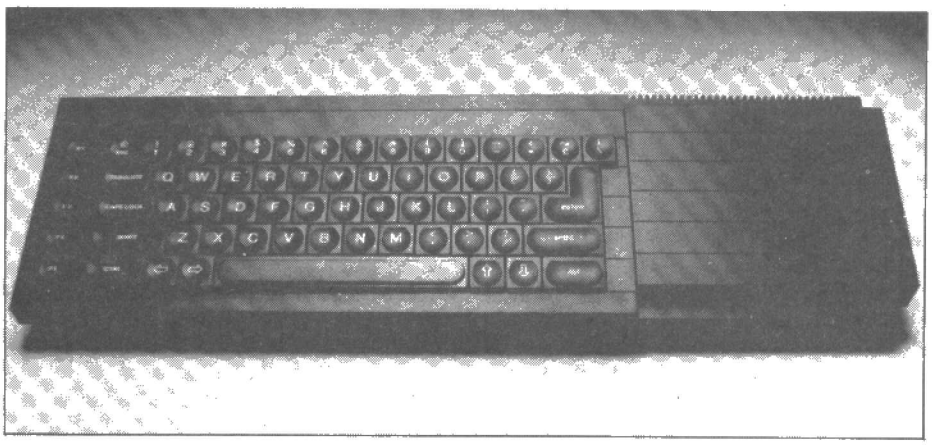
KEY keynumber, "function key string"

Where keynumber is the number of the function key we're programming, from 1 to 5, and function key string is any string expression of between 0 (the null string, which removes a definition) and 80 characters. To cause a line to be executed, we must append a new line to the key string, so if we want to program key F1 to produce a directory of floppy disk 1, we type

KEY 1,'dir flp1_'&chr$(10)

Now every time we hit F1 a directory of flp1_ is produced on the screen.

One more thing we'll need to add is a switch to turn the definitions off. If you hit CTRL–CAPS LOCK at any time, the definition state is toggled (ie if it is on, it will go off; if it off, it will go on). We need to do this to avoid confusing Quill, Ed and so on. Of course, there's absolutely no reason why we should append a command to a function key, giving us instant access to Quill (etc) macros. If we were using Metacomco's screen editor, for example,



and wanted to replace every occurrence of 'textual' with 'froglet', we could program F3 beforehand with the appropriate command sequence.

The same applies to the Psion packages (it certainly works with version 2, anyway). Instant macros!

Now for the code which is shown in **Listing 1.** The assembled code is loaded into the resident procedure area and then called to link in the two routines:

a=respr(704):lbytes flp1_qkey,a:call a

The code which is actually called is that starting at the START label. It simply calls BP_INIT to link in the SuperBasic routine, and then MT_LPOLL to link in the interrupt routine.

The interrupt routine is at KEY_INT and first gets the address of the current keyboard queue into A2. The address of the next position to write to is then placed in A1 and a check is made to see if this is at the start of the queue (remember that the buffer is *circular!!*). If it is, we alter A1 to point to the end of the queue, and in either case we subtract one to point to the last character written into the buffer. A DBRA loop scans a table of key codes, and if this coincides with the character read in we jump to KEYCHANGE. Notice how this code is preceded by a check to see if CTRL–CAPS LOCK is pressed, when the lower bit of FUNCSON is toggled with EDR.

At KEYCHANGE we calculate the address of the function key's string and enter a DBRA loop with repeated calls to IO_QIN to insert the string into the queue. If this fails at any point, the buffer must be full so we leave straight away.

The KEYS procedure isn't exactly complicated and is the sort of stuff we've met before in this magazine. It takes two string arguments, converts the first to a number and inserts the second at the requisite place in the KEYDEFS buffer.

RELAY BOARD & OPTICAL SENSOR

This project complements the I/O board described in our January issue. Design by Chris Walcot.

In the January '85 issue of *E&CM* we explained how to build a simple input/output port for the Dragon 32. On its own this was a very limited device, but by using additional hardware you can expand its capabilities: this article describes a relay board and an optical line sensor which does just that.

The relay board

The relay board can control high powered devices. It consists of eight relays connected between the data lines and the +12V supply from the Dragon. In theory, the 6821 PIA (peripheral interface adaptor) should handle only +5V, but the relay limits the current flow to keep it within a safe level. Our choice of relay was a 12V, SPST, ultra miniature device; this is both compact and reliable and needs no diode to prevent back emf (electromagnetic force). The circuit diagram is shown in **Figure 1.**

The relay is built on a piece of PCB. 0.1 inch matrix IDC cable and sockets are used to connect the PCB to the interface. The relay contacts are connected to 4mm single pole sockets, mounted on the surface of the plastic case.

First connect the IDC cable to the 12 way IDC socket. This is not too difficult but is made easier with an insertion tool. Once that is completed, strip the other end of the cable and solder it in position to the PCB, then insert and solder the wire links. The relays should be attached carefully because they are the most easily damaged component in the circuit. Next the leads (approximately 6cm long) should be soldered to the 4mm socket at one end.

The lid of the box should be drilled with an 8mm drill for the 16 socket holes. A small lip should be cut in the side of the box to allow the cable to pass through. Finally, attach the sockets in place, solder the other end of the leads to the PCB and firmly secure the lid.

Testing

Once the board has been checked thoroughly plug it into the Input/Output board (WHEN THE COMPUTER IS SWITCHED OFF) described in the January issue of *E&CM.* and enter the software which was given in that article.

It is then necessary to make up a simple indicator circuit consisting of a LED in series with a current limiting resistor. Program the port you are using to output data, and make all the outputs high (1). Place the 4mm plugs in turn in each pair of sockets on the relay board – each time the LED should light up. Now set all the outputs to low (0), and in each position the LED should be off. Once that is completed try different combinations of highs and lows. If any results are not as they should be then turn off the computer for further investigation.

How it works

There are two sockets for each relay (which acts as a switch). When the relay is activated the switch is closed (the two sockets are connected together); when the relay is not energised the switch is open (the two sockets are not connected together).

Any device, where a switch is needed, can be attached as long as it does not draw more than 2A at 24V DC.

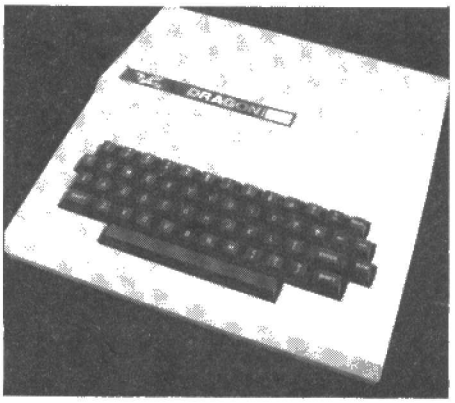
The optical line sensor

The optical line sensor can be used for a wide range of tasks from checking paper has not run out in a printer to allowing a robot to follow a line. It works on the simple principle of bouncing light off a surface. If the surface is black then very little light will be reflected. It it is a white or metallic surface then a large amount of the beam will bounce back off the object.

An Infra Red Light Emitting Diode (when 'on' it gives off infra red light which is invisible to the human eye) is used for the light source. For the detector an Infra Red Photo Transistor (a special type of transistor which allows a current to flow when infra red light is directed onto it) was chosen. These two devices can be bought in one plastic package. For this circuit a Miniature Reflective Optical switch is used. **Figure 2** shows the circuit diagram.

Construction

The IDC cable should be made up and stripped at one end (as described for the relay board) and soldered to a piece of Vero board no smaller than 12 x 12 holes. One track should be cut, and the 330 ohm



resistor soldered across it. Next, solder a lead to pin number 1 of the opto-switch

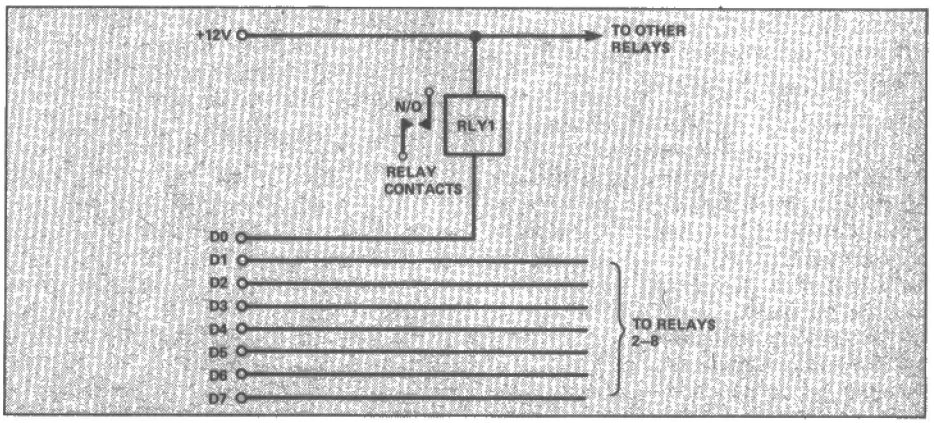
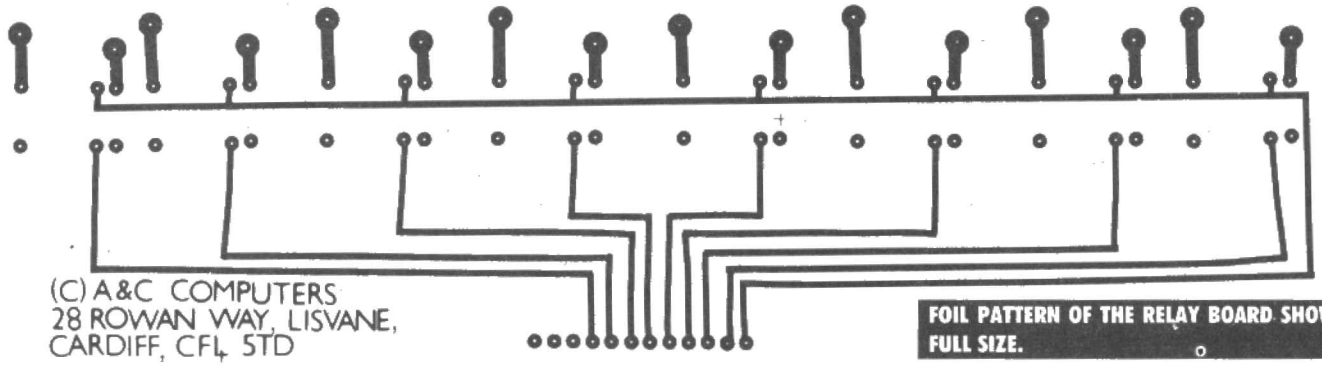


Figure 1. Only one relay is shown.



FOIL PATTERN OF THE RELAY BOARD SHOWN FULL SIZE.

and to the Data Line. Pin number 2 should be connected to the 0 volts and pin number 3 to the resistor. The 4th pin should be connected to the 0 volts supply.

Testing

As previously explained, plug the lead into the I/O board's connector. Set up the port for input using the software. Place the opto-switch about 0.4mm above a white surface, then press 'F' on the keyboard and a '1' should appear for bit 8 of that port. If the test is carried out again, but with a white surface, then a '0' should appear for bit 8.

This device could be given a longer range with the use of a different opto-switch, but some extra circuitry might be needed.

PARTS LIST

RELAY BOARD

8 x 12V SUB MIN RELAYS
1 x 12 WAY IDC CABLE
1 x 12 WAY IDC SOCKET
8 x 4mm RED SOCKETS
8 x 4mm BLACK SOCKETS
1 x PVC CASE

OPTICAL LINE SENSOR

1 x MIN OPTO SWITCH
1 x 300 ohm RESISTOR
1 x 12 WAY IDC CABLE
1 x 12 WAY IDC SOCKET
1 x 10 by 10 VERO BOARD

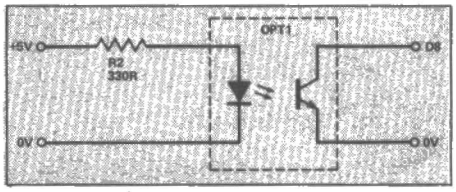


Figure 2. Showing connection of optical switch.

All devices described in this and the previous article are available full built and test from A&C Computers.

I/O board	£20.00
Relay board	£26.50
Optical line sensor	£10.90

Any parts you have difficulty in obtaining can also be supplied. *A&C Computers, 28 Rowan Way, Lisvane, Cardiff CF4 5TD. Tel: 0222 756653.*

DRAGON COLOUR KILLER

Peter Luke shows how some minor Dragon surgery can lead to a dramatic improvement in the computer's video display.

The Dragon computer provides an ideal, low cost route to a FLEX or OS9 development system. In this environment, the computer will most likely be used with a monochrome monitor set, the Dragon providing a suitable composite video signal. Anyone who has used a Dragon with such a monitor will be aware of one serious draw back with this arrangement – the dreaded dot crawl. This irritating effect, in which the text displayed on the screen is overlayed with a random collection of crawling dots, is caused by the 'colour' information that is present in the Dragon's composite video signal. The chrominance signal is at 4.43MHz, right in the middle of the bandwidth of many monitors.

There are a number of ways of overcoming this problem. A low pass filter cutting of a frequency less than 4MHz would work but this would have an adverse effect on the resolution of the display. A notch filter would be a better idea, but this too could have implications on the ultimate resolution and would also cost money to build. Fortunately there is a way to cure the effect that will not cost users a penny.

The colour subcarrier of the Dragon's video signal is provided by an LM1889 video modulator IC. This clever piece of silicon generates the PAL colour subcarrier from the R-Y and B-Y signals produced by the Dragon (don't worry if you're not familiar with the jargon of colour TV engineers, as you'll see it is not necessary to have an intimate knowledge of the way in which a colour display is generated to make our modification). The point is that the LM1889 adds the subcarrier to the video signal, the video signal itself does not pass through the IC. To get rid of the dots then, all we need do is remove the offending IC.

On some Dragons the IC is socketed and the job of removing it is easy. The '1889 is located toward the rear of the PCB just right of centre. It is clearly marked and should not be difficult to locate. Take care not to bend any of its 'legs' when extracting it as you may well want to restore the colour capability of the Dragon at some future date. Needless to say this operation should be performed with the computer switched off and, preferably, the power supply disconnected.

If your computer features an LM1889 that is soldered in things are not quite as easy. The best way is to solder a large capacitor (greater than 100nF) across the crystal located near the IC. The crystal is in a tin case and will be marked with the numbers 4.43 somewhere on its body. To restore colour simply remove the capacitor.

The improvement in the quality of the display is dramatic, and the fuzzy text that so many users become resigned to can be banished for ever.

One note of caution, removing the Dragon's cover to make this alteration will invalidate any guarantee that many be in force.

ON/BRK/GOTO

Joe Pritchard's Spectrum control program

How does it work?

On calling address 60000, the address of the main routine (BRK), is stored in the memory location "VECTOR". Interrupts are then disabled and the interrupt mode of the Z-80 is changed so that whenever the normal Spectrum interrupt occurs – once every fiftieth of a second – the routine whose address is held in "VECTOR" is entered and executed.

The main routine functions by examining the contents of 23610, which indicates whether an error has been detected by the system. If this address contains a value of #14, then it indicates that the Break error has been detected. But if this hasn't happened, we simply clear things up and return to the BASIC interpreter by making a jump to the keyboard routine at address #0038. If a Break error has been detected, then the routine "DOIT" is entered. Lines 230 to 260 of the program call a ROM routine that creates a 'space' into which we insert, in Lines 270 to 280, the BASIC token for "Continue". On return to the BASIC interpreter, this will be executed as if we'd typed it in ourselves.

However, we've not finished yet. Lines 290 and 300 restore the stack, by first removing the AF register that was pushed on to the stack, and then by removing the return address for the interrupt, which we don't need. The line number to which we want control to pass is loaded into the System Variable OLDPPC at lines 310 to 320. OLDPCC specifies the line number that is to be jumped to when CONTINUE is executed. Finally, in Lines 350 to 360 we set OSPCC to 1, to indicate that the BASIC interpreter is to execute statement 1 in the line held in OLDPCC. Interrupts are then re-enabled, and a jump is made to the point in the BASIC Interpreter that will allow BASIC to execute the "CONTINUE" command and go on with the program.

In Alan Went's Spectrum Wordprocessor program, published in this magazine in October 1984, there was a small piece of machine code that disabled the Spectrum Break key. In this article, I describe a machine code routine that causes the BASIC program to jump to a particular line number when the CAPS-SHIFT and SPACE keys are pressed together. This is a useful feature in programs that are to be used by "non computer" people, as it provides them with a panic button that can be used to return to a certain place in a program so that they can regain control.

In the listing below, the machine code causes a jump to line number 9000, but this can be altered as we'll later see.

```
10          ORG   60000
20 VECTOR   EQU   #FEFF
30 BRKON    LD    HL,BRK
40          LD    (VECTOR),HL
50          DI
60          LD    A,#FE
70          LD    I,A
80          IM    2
90          EI
100         RET
110 BRK     DI
120         PUSH  AF
130         PUSH  HL
140         LD    A,(23610)
150         CP    #14
160         JR    Z,DOIT
170         LD    HL,23692
180         SET   /,(HL)
190         POP   HL
200         POP   AF
210         EI
220         JP    #0038
230 DOIT    POP   HL
240         LD    HL,(23641)
250         CALL  #1652
260         INC   HL
270         LD    A,232
280         LD    (HL),A
290         POP   AF
300         POP   HL
310         LD    HL,9000
            ; the line
            number to be
            jumped to
320         LD    (23662),HL
330         LD    A,1
340         LD    (23664),A
350         LD    A,255
360         LD    (23610),A
370         EI
380         JP    #12E2
```

This program is activated by a call to "BRKON", which I have placed at address 6000. Wherever you put the code, ensure that the two memory locations at #FEFF are not overwritten, as these are crucial to the operation of the program. Obviously, once active there should be a line 9000 in the program.

AN ENTERPRISING INTERFACE

Paul Beverley turns his attention to the new Enterprise computer. The article expands on the details provided in the computer's programmer's guide.

First impressions of the Enterprise indicate that at least a similar design philosophy to the BBC micro has been applied. This introductory article looks briefly at the system as a whole and then in particular at what sort of interfacing facilities it has. Reasonably experienced "interfacers" will appreciate the inclusion of some interfacing details that do not appear in the Enterprise Programmers' Guide. Much of this information has been provided by Mr. Steve Groves of Enterprise Technical Services, for which, many thanks. Would that Acorn had been so co-operative at such an early stage!

Another aspect which will be looked at is how to link the Enterprise up to a Microvitec RGB monitor. When this was initially tested the range of colours claimed by Enterprise was not being given. This was not the fault of the computer though. The monitor was set up for the digital TTL voltages produced by the BBC micro, but the signals from the Enterprise need a linear input to get the various levels of colour. Fortunately the monitor can be altered by making some simple link changes to configure it as a linear monitor. So in case you have the same problem details of these link changes will be included.

"... The hardware of the Enterprise is more advanced than the BBC – but then it should be ..."

The hardware

The hardware of the Enterprise is clearly more advanced than the BBC, but then it should be, with the benefit of three years of technical advance. It consists basically of a Z80 processor, 32K ROM and 64K RAM on the main PCB, plus a 16K BASIC ROM in a plug-in cartridge. It has two custom-designed chips – NICK and DAVE, both 68 pin packages. NICK's job seems to be to look after the RAM and act as a CRT controller, while DAVE generates the sound and takes care of various other input/output and internal control functions.

As regards software, Enterprise have continued the moves started by the BBC microcomputer by using structured BASIC with facilities such as multi-line IF, DO WHILE, and CASE statements. There is also a built-in word-processor and some of the word processing facilities seem to be used by BASIC which features a full-screen editor. More than one screenful may be added since you can scroll downwards to some extent as well as upwards, thus avoiding that terrible sense of frustration that occurs when, as you list your program, the line you want to see just disappears off the top of the screen. To retreive it simply press SHIFT-cursor-up to move the cursor to the top of the visible screen and then continue to move up so that the screen scrolls down until the missing line re-appears at the top of the screen!

Another similarity with the BBC is the idea of having the operating system (EXOS – Enterprise Xpandable Operating System) entirely separate from the high level languages. This is the theory, but just how it words in practice is not yet clear. BASIC is said to be contained in the 16K ROM cartridge, but if the bits of text that appear on the chip on the main PCB marked as EXOS (a 27256 – 32K EPROM!) are anything to go by, it seems to contain some of the coding for the BASIC interpreter as well as the operating system and the word processor software.

Interfacing facilities

An examination of the basic functions of each of the connectors on the computer will be helpful. Almost all the interfaces use double-sided edge connectors with 2.54mm (1/10 inch) spacing. The exceptions to this are the cassette input/output and remotes, which are jack sockets, and the TV output which is a phono socket.

There is very little to say about the UHF output except to note that it is picture only, and does not carry any sound signal. Enterprise seem (hopefully) to have learned from the troubles experienced with almost all other home computers with cassette interfaces. There is on-screen indication of the in-coming tone and also, a proportion of the signal is passed to the audio section of the computer to give audible confirmation that loading is taking place. I certainly had no problems with the tape-recorder I tried. There are two remote switches, one associated with each of the cassette input and output lines. This is presumably to allow individual control of two recorders for file manipulation and transfer from cassette to cassette.

From an interfacer's viewpoint the fact that two remote switch relays can, when not being used for the cassette recorder, be individually controlled from software by saying, for example, SET REM1 ON, or TOGGLE REM2. The relays can be used directly to control low power DC devices, or by using them to switch larger DC operated relays, they could control somewhat larger loads.

The pin connections for all the edge connectors are given in **Figures 1 to 7.** The order in which they are given is in a clockwise direction, starting from the left-hand side of the keyboard.

1) Cartridge Connector

This is a 32-way edge connector (ie 16 + 16), but unlike all the others, it is a socket. This is so that the cartridge can just consist of a PCB with a ROM or ROMs on it, and the expensive socket is contained within the computer. This socket is used for the Intelligent Software BASIC which comes with the machine.

2) The Monitor Connector

Apart from the red, green, blue and composite sync signals needed for an RGB monitor, there is a composite signal for a black and white monitor made up by summing the RGB signals. Also on this connector are outputs from the two sound channels so that you can apply them to the auxiliary inputs of a stereo amplifier. The line marked "Peritel switch" is just an output from the +12V line through a 1K resistor and is apparently for certain continetal TVs which will switch automatically to a monitor mode when they receive this signal.

3) The Serial/Net Connector

This has serial data input and output lines and also handshaking input and output lines. All the signals are at 0-12 levels, so that if you want to link into any RS232 or RS423 systems the appropriate line drivers and receivers will need to be added.

4) Printer Connector

This appears to be a standard Centronics type of parallel interface though it is on an edge connector rather than the more standard Centronics or IDC connectors. The outputs are buffered by a 74LS244 which suggests the possibility of being able to use it as a digital output port.

5 & 6) The Control Connectors

Each of these has basically the same set of connections – five output lines, three input lines, ground and +5 volts. The three input lines are common to both connectors, and the ten output lines (five on each connector) are the same lines that the computer uses to scan the keyboard. The obvious use of these interfaces is for joysticks which can be read with the JOY(N) function where N is 1 or 2 for the appropriate connector.

7) The Expansion Connector

This has just about every connection you could think of for adding on all sorts of interesting hardware. The majority of the signals consist of the address, data and control lines of the Z80 microprocessor, and the rest are various clock, power supply and control signals as well as two sound inputs so that the external hardware can add to the audio signals that the DAVE chip produces. The power supply line is the unregulated +9 volts that comes from the transformer.

The clock signals, apart from the Z80 clock (4MHz), are 1MHz, 8MHz and 14MHz. There are six extra (paged) address lines suggesting a memory addressing capability of 4 Mbytes. Then there are the vertical and horizontal sync outputs as well as four inputs labelled EC0 to EC3 which can be used to give the external hardware direct control of the colour signal generation. This is enabled by the EXTC line which is negative active and, if unused, pulls high thereby disabling the external colour control.

Using a Microvitec Colour Monitor

The connections for the R, G, B and sync outputs from the computer are given in **Figure 2,** and the connections to the socket on the monitor (Microvitec models 1431, 1441 or 1451) are given in **Figure 9a.** We have not yet managed to find a source of the proper edge connectors; a sawn up piece of an old, somewhat larger edge connector socket may well have to suffice! If you make up a suitable cable and connect it to the monitor you may find that the colours are not the same as those of the television picture. The reason is that the monitor is expecting digital (on/off) colour signals whereas the computer is producing colour output voltages varying between 0 and about 4 volts. However, the Microvitec monitors have internal links which allow them to respond correctly to these linear colour signals.

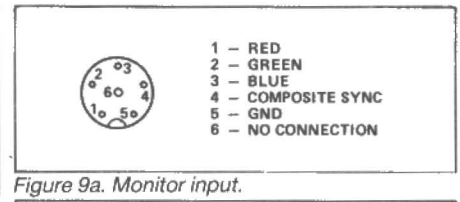
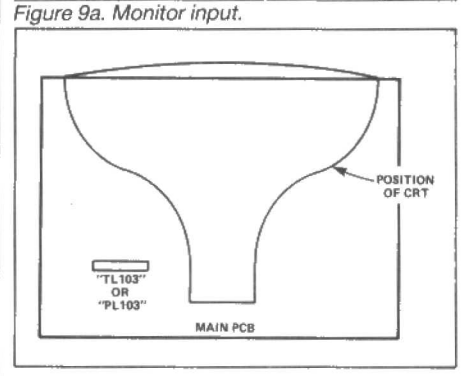
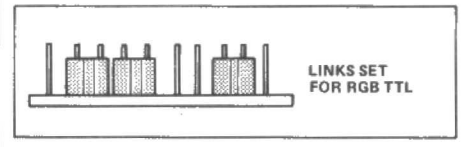


Figure 9a. Monitor input.





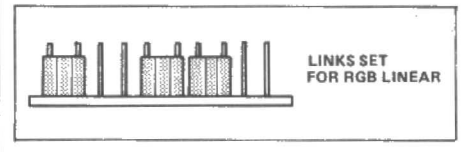


Figure 9b, c, and d. Location of links within monitor and details of the changes for linear RGB operation.

Figure 1. Cartridge Connector

(Side A is the under-side, B the top-side. The connections are numbered from left to right looking into the socket ie B1 is on top, nearest the back of the computer.)

Pin	Signal	Pin	Signal
A1	GND	B1	+5V
A2	D3	B2	D4
A3	D2	B3	/RD-CART
A4	D5	B4	/CART
A5	D6	B5	A15
A6	D1	B6	A14
A7	D0	B7	A13
A8	D7	B8	/WR
A9	A1	B9	/RD
A10	A2	B10	A12
A11	A5	B11	A11
A12	A4	B12	A10
A13	A3	B13	A9
A14	A0	B14	A8
A15	A6	B15	A7
A16	GND	B16	+5V

Figure 2. Monitor Connector

(Side A is the under-side, B the top-side. The connections are numbered from left to right looking into the edge connector from the back of the computer ie B1 is on top, furthest away from the power inlet and UHF output connectors.)

Pin	Signal	Pin	Signal
A1	GREEN	B1	N/C
A2	GND	B2	GND
A3	BLACK/WHITE	B3	BLUE
A4	/HSYNC	B4	RED
A5	/VSYNC	B5	/CSYNC
A6	—	B6	(PERITEL SWITCH)
A7	LH AUDIO-OUT	B7	RH AUDIO-OUT

Figure 3. Serial/Net Connector

(Numbering is in the same orientation as the monitor connector.)

Pin	Signal	Pin	Signal
A1	REF	B1	GND
A2	—	B2	—
A3	STATUS OUT	B3	DATA OUT
A4	STATUS IN	B4	DATA IN

Figure 4. Printer Connector

Pin	Signal	Pin	Signal
A1	GND	B1	GND
A2	/STB	B2	/RDY
A3	D3	B3	D4
A4	—	B4	—
A5	D2	B5	D5
A6	D1	B6	D6
A7	D0	B7	D7

Figure 5. Control Connector 1

Pin	Signal	Pin	Signal
A1	KBJ	B1	GND
A2	KBL	B2	KBK
A3	—	B3	—
A4	KB4	B4	+5V
A5	KB2	B5	KB3
A6	KB0	B6	KB1

Figure 6. Control Connector 2

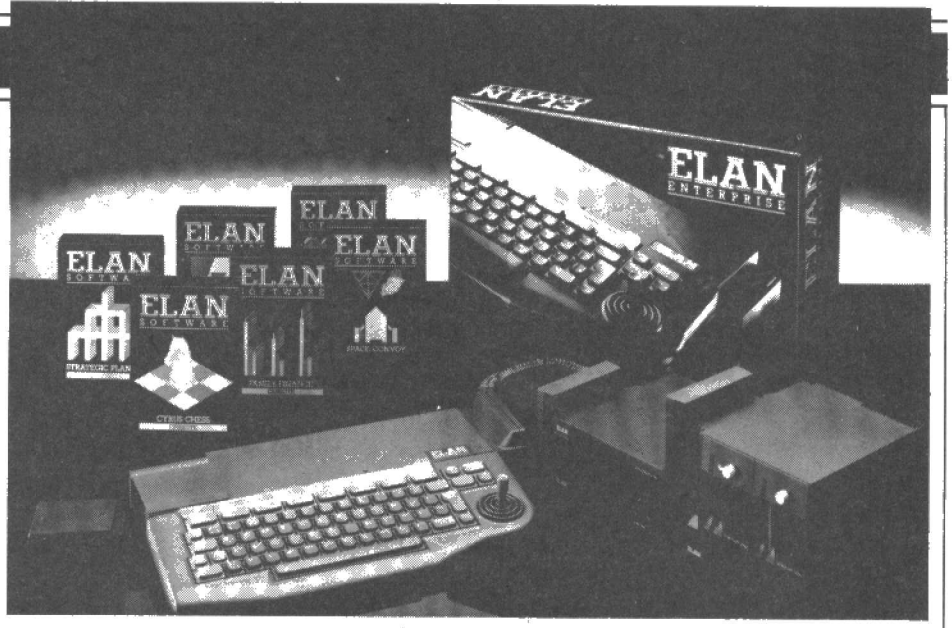
Pin	Signal	Pin	Signal
A1	KBJ	B1	GND
A2	KBL	B2	KBK
A3	—	B3	—
A4	KB9	B4	+5V
A5	KB7	B5	KB8
A6	KB5	B6	KB6

Microvitec Monitor Link Changes

[WARNING. Monitors contain high voltage circuits which are potentially LETHAL, you should NEVER operate the monitor when the case is open, and you should remember that the voltages may not decrease to a safe level until some time after the unit has been switched off. Therefore it is best not to do these modifications until the monitor has been unused and switched off for several hours.]

In order to make the necessary link changes you will need to get inside the case. With the metal-cased units this just means unscrewing the back, but the plastic-cased units need to be completely dismantled. Having done this, locate the main PCB which is horizontal. The link selection connector is located, as illustrated in **Figure 9b,** towards the front left of the PCB as you look into the case from the rear and it is marked as either "TL103" or "PL103". It consists of a set of 10 Molex pins with three links or jumpers. Each of the three links needs moving (one each for R, G and B).

"... Linear RGB operation displays the full range of colours ..."



The original positions are shown in **Figure 9c** and the new positions in **Figure 9d.** When the links have been changed, screw the case back together again and connect up to the Enterprise. Note that in the linear mode, the "contrast" control on the monitor is inoperative as it is only intended to be used in the digital or TTL mode.

In the digital mode, the only colours available are red, green and blue, and their combinations, cyan, magenta, yellow and white. The simplest way therefore to test whether the monitor is working properly in the linear mode is to use the Enterprise logo which appears at switch-on. In order to check the colours without them changing all the time, you can press the HOLD key which freezes the display. The characters of this logo go through all the colour levels, so by repeatedly pressing HOLD to start and stop the display you can check the different colour levels. However, if you want to see all the colours on the screen at the same time, you can use the program on pages 93 and 94 in the Programmers Guide.

Figure 7. Expansion Connector

(Numbering is in the same orientation as the monitor connector ie B1 is on the top-side nearest to the front of the computer.)

Pin	Signal	Pin	Signal
A1	RH AUDIO-IN	B1	LH AUDIO-IN
A2	/RF8H	B2	/WR
A3	/RD	B3	/IORQ
A4	N/C	B4	N/C
A5	/MREQ	B5	/NMI
A6	A8	B6	A9
A7	A10	B7	A11
A8	A12	B8	A13
A9	A14	B9	A15
A10	A0	B10	A1
A11	A2	B11	A3
A12	A4	B12	A5
A13	A6	B13	A7
A14	D0	B14	D1
A15	D2	B15	D3
A16	D4	B16	D5
A17	D6	B17	D7
A18	/RESET	B18	/INT
A19	/WAIT	B19	GND
A20	/M1	B20	GND
A21	1MHz	B21	GND
A22	PHI	B22	GND
A23	8MHz	B23	GND
A24	EC0	B24	EC1
A25	EC2	B25	EC3
A26	/EXTC	B26	A16
A27	A17	B27	A18
A28	A19	B28	A20
A29	A21	B29	14MHz
A30	GND	B30	/VSYNC
A31	GND	B31	N/C
A32	/HSYNC	B32	GND
A33	+9V	B33	+9V

Figure 8. Words occurring in the EXOS SYSTEM ROM and the BASIC ROM.

a) EXOS ROM (32K)

EXOS 2.0
TESTING ERROR INTERNAL CHECKSUM ERRORS EXOS_ROM TEST_ROM TAPE
Written by: Mr1 BT NMV GHN CGE AEL

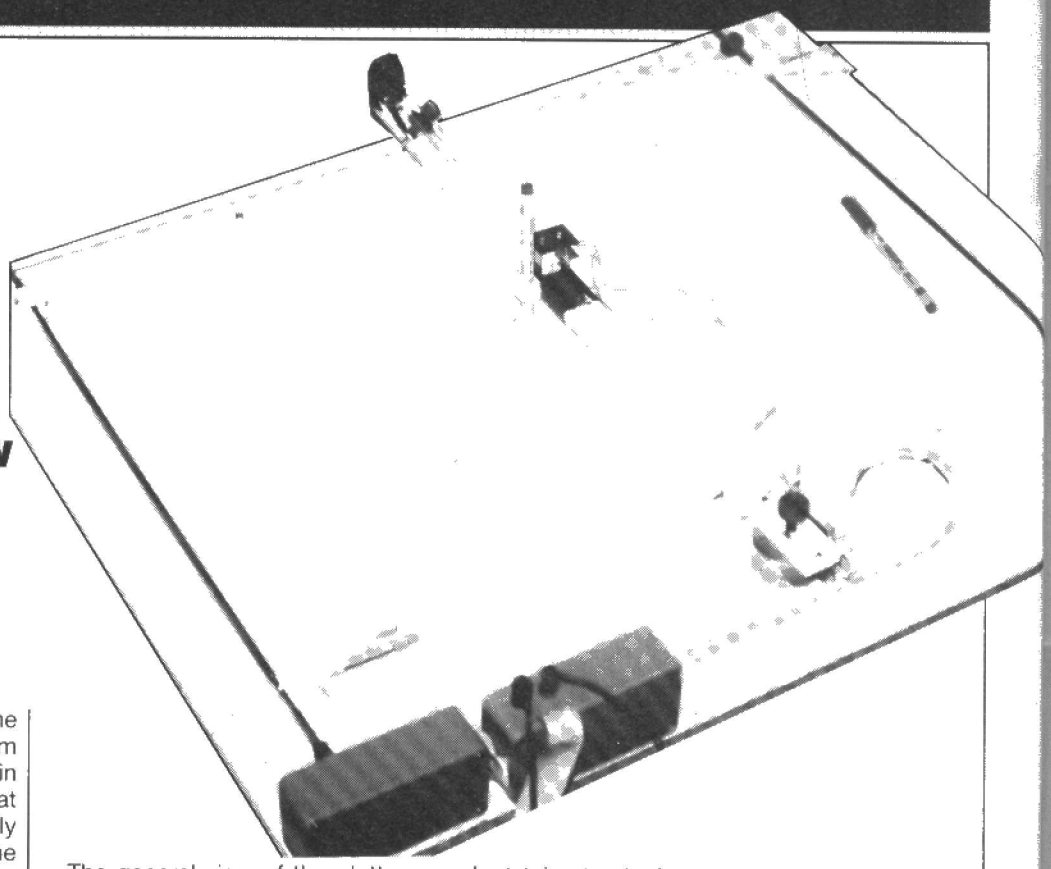
HELP LOAD Error. Re-starting system.
VIDEO SOUND CAPS SHIFT HOLD ALT KEYBOARD EDITOR PRINTER

Enterprise IS-BASIC version 2.0 1984 Intelligent Software Ltd
ABS ACOS ANGLE ASIN ATN BIN BLACK BLUE CHR$ COS COSH COT CSC CYAN DATA$ DEG EPS EXLINE EXP EXSTRING$ EXTYPE FREE GREEN HEX IN INKEY$ INF INT IP JOY LBOUND LCASE$ LEN LOG LOG2 LOG10 LTRIM$ MAGENTA MAX MAXLEN MIN MOD ORD PEEK SPEEK PI POS RAD RED REM RGB ROUND RTRIM$ SEC SIN SINH SIZE SGN STR$ SQR TAN TANH TIME$ TRUNCATE UBOUND UCASE$ USR VAL VER$ VERNUM WHITE YELLOW WORD$
TYPE TOGGLE_SPEAKER DISPLAY_GRAPHICS DISPLAY_TEXT TOGGLE_REM2 RENUMBER LLIST CONTINUE INFO TOGGLE_KEY_CLICK GRAPHICS TEXT TOGGLE_REM1 AUTO LIST START ATTRIBUTES AT BAUD BORDER BEAM BIAS BUFFER CHANNEL CHARACTER CLICK CODE COLOR COLOUR CURSOR DELAY DEFAULT DOWN EDITOR FAST FIST FKEY FORMAT INK INTERRUPT KEY LAST LINE LEVEL MACHINE MODE NET NUMBER ON OFF PALETTE PAPER PROTECT RATE REM1 REM2 SAVE SCROLL SERIAL SOUND SPEAKER STEP STOP STATUS STYLE TAB TAPE TIMER TO UP USING VIDEO
GNIDAOL GNIVAS GNIHCRAES DNUOF ESUAP

(Word processor prompts also appear in the EXOS ROM.)

DIY PLOTTER

J. Pilkington describes the construction of a low cost, high performance X-Y plotter.



While a printer is likely to remain the prime means of producing hard copy output from a computer system, interest in CAD and in the LOGO computer language means that a plotter is becoming an increasingly popular computer add-on. But while the cost of commercial plotters remains at the present high level few individual computer users will be able to afford this luxury.

The alternative is to take the DIY route, which brings the price of a plotter down to a level which even the most cash starved micro user can afford.

The design brief of the *E&CM* plotter included the requirement that the finished article should be fabricated from low cost components and that it should offer a potential performance similar to that of commercial designs.

The finished design uses two stepper motors, each motor moving the pen in 0.2mm steps both in the X and Y planes. This gives more than adequate resolution; the plotter, when drawing geometric shapes such as squares, triangles etc. manages to end up at the starting point!

The design as presented here, although offering excellent performance, is still capable of improvement. Therefore rather than giving a detailed description of the construction of the plotter, only broad guidelines will be provided. These will contain sufficient detail for readers to build the plotter as per the prototype yet will allow for improvement on the design it doesn't have to be a slavish copy of the original.

"... Experiment with CAD software ..."

Building blocks

This month we will show photographs of all the major sections of the plotter togehter with brief descriptions of their construction. Next month we will provide detailed drawings of the mechanisms together with details of the circuit necessary to interface the plotter to the BBC micro.

The general view of the plotter reveals that the basic mechanism is built from 'Swish' plastic curtain track glued to a base board formed from melamine-faced chip board. Extensive use is also made of the sliding fittings that are supplied with the track in order to save money. The curtain track is an ideal component for the plotter providing a smooth guide to the Y arm.

The pen can be moved in four fundamental directions or in combinations of these directions. It can be raised or lowered and there is a home reference point that can be used to confirm that the pen is on course at regular intervals during use.

"... The plotter is built from low cost components yet offers a performance similar to commercial designs ..."

The pen carriage

This assembly must be able to slide easily, have tie points at the front and back and have a bracket to hold the pen lifting solenoid.

The best way to fabricate the pen carriage is to first make a paper template and then use this to form the final component from a thin sheet of aluminium. The Radio Spares solenoid used in the prototype had two holes drilled and tapped in its frame and it was relatively easy to bolt this to the frame.

The pen holder

The holder was fashioned from a small length of curtain track cut down to a block with a pen drilled through it to take the pen used. The nylon track from which the curtain track is manufactured is elastic enough to give a good grip on the pen without the need to ream it out to the exact diameter. It is important though to make sure that the pen doesn't move too much in its holder as the assembly changes its direction of movement.

The pen holder is attached to the solenoid lever by the simple expedient of making a saw cut down one side of the block and using a blob of glue to attach this to the arm. This arrangement has proved perfectly satisfactory during use.

Tensioning the cord

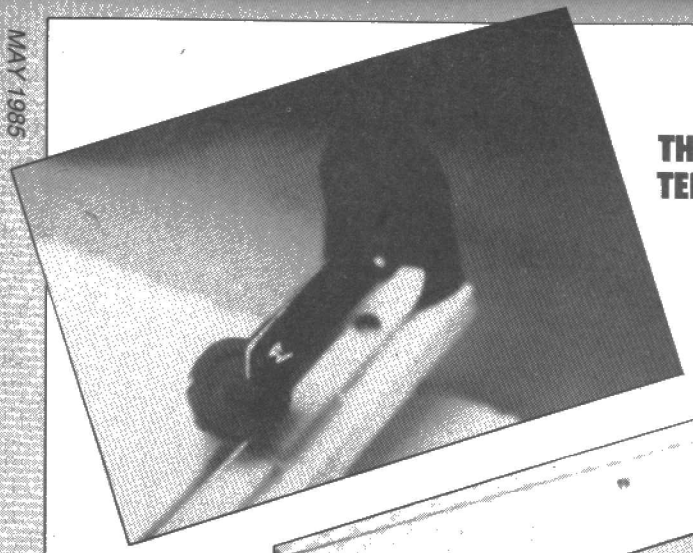
The photographs show that the prototype design used elastic bands to tension the pulleys on the Y axis arm and a long threaded bolt on the X axis. In retrospect, an elastic band would probably have done the job for both axes.

Next month – *detailed diagrams of the major plotter components, details of the interface electronics and ideas for extending the design.*

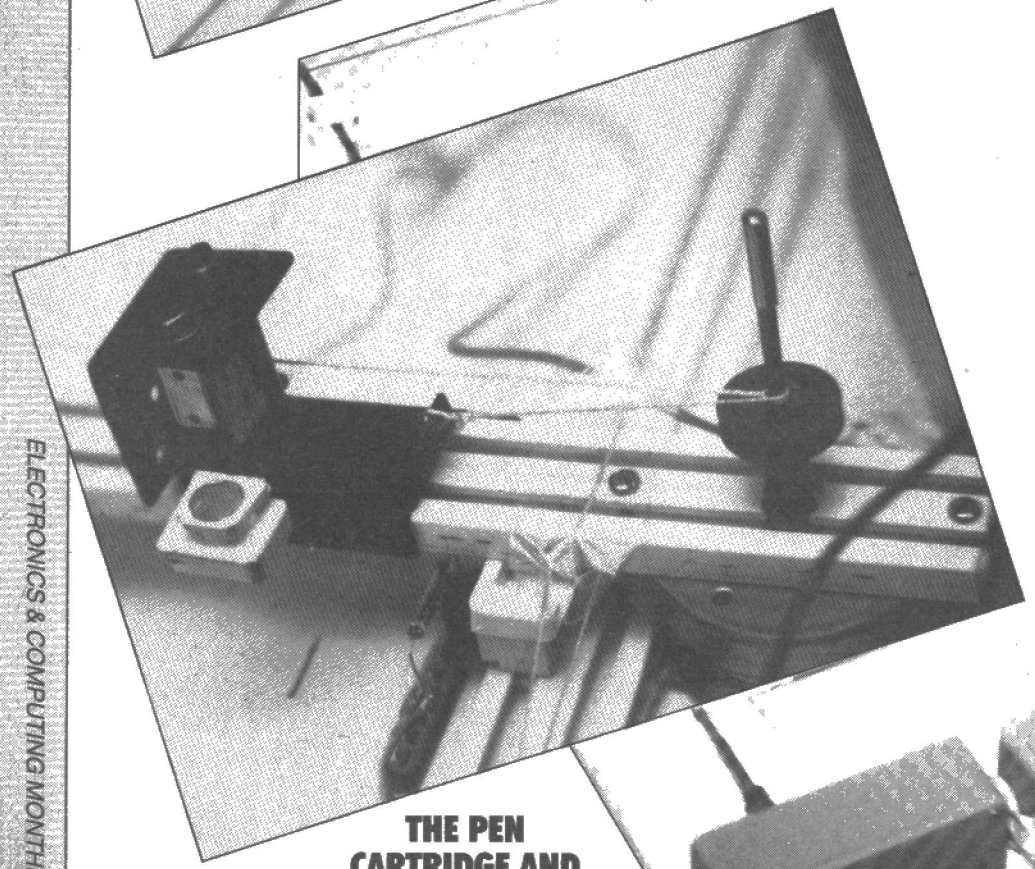
About the author – *John Pilkington is Head of the Faculty of Design and Technology at St. Andrews Church of England School in Croydon. He designed and built the plotter on a five day workshop at an MEP centre.*

The author wishes to thank Mike sharp of Capital Region MicroElectronics for the design of the stepper motor interface and the sample program (to be given next month).

PLOTTER CONSTRUCTION



THE Y AXIS TENSIONER



THE PEN CARTRIDGE AND HOLDER

THE X AXIS TENSIONER

THE DIAGRAM DIAGNOSED

Mike James concludes his voyage around the BBC micro's circuit diagram with a description of the various I/O facilities offered by the computer.

The analogue port provided by the A to D converter chip (IC73) is very simple. Its data and control registers are connected to the system data bus and its address is decoded by IC24 as already described. The end of conversion signal is connected to CB1 of the system VIA. Thus conversion consists of writing data to the control register, waiting for the end of conversion signal and then reading the data. The only point worth noting is the need to derive separate read and write strobes for this chip, a task undertaken by IC77 in the disk controller circuit. Problems and solutions to the instability of the voltage reference provided by D6 to D8 have already been described in *E&CM* a number of times.

The sound chip, IC18, is interfaced to the system via the slow data bus provided by the A side of the system VIA. (Notice that this means that the sound chip doesn't appear in the BBC Micro's memory map.) The sound chip is selected by the addressable latch as already described. Analogue processing is provided by IC17 and IC19. Notice that there is a volume control (VR1) just before the final stage of amplification.

Cassette and serial I/O

The cassette and serial interface is provided mainly by the serial ULA IC7 and a standard 6850 ACIA IC4. The ACIA is connected to the system data bus and is selected by IC26 as described earlier. What is interesting about this configuration is the way that the ULA takes the signals from the ACIA and sends them to provide a cassette interface. The serial ULA contains a baud rate generator, tone detector, data separator and a sine wave synthesiser. This combination provides a sophisticated serial interface (with software programmable baud rates) and a very reliable cassette interface. When writing to the cassette the serial data drom the ACIA simply controls the frequency produced by the sine wave generator in the serial ULA. When reading data from the cassette, the data separator changes the audio tones into a serial data stream and a clock signal. The serial data stream is applied to the serial input of the ACIA and the clock pulses to the receive clock input. In this way the data recovery from the cassette is very tolerant of speed fluctuations. The analogue side of the circuit is composed of IC35 and two transistors Q1 and Q2. The cassette motor is controlled by a small relay driven by Q3. The signal side of the RS423 interface is provided by IC74 and IC75.

The user VIA & 1MHz bus

The second VIA in the system is dedicated to providing an eight-bit user port and a standard Centronics printer interface. It is connected to the system data bus and it is selected by IC24. The printer port is formed from the A side data lines, buffered by IC70 and CA1 and CA2 for handshaking. Notice that CA2 is buffered by IC27 and a transistor (Q11) to increase its current drive capability. The user port is provided by all of the B side lines PB0 to PB7 and CB1 and CB2. Notice that the user port is not buffered in any way and therefore has limited current drive. On the plus side the lack of buffering means that any data line in the user port can be used for input or output as desired.

The 1MHz bus provides a fully buffered (IC71) subset (A0 to A7) of the system address bus and a fully buffered (IC72) copy of the system data bus. Two additional select lines are provided by IC20 for the Fred and Jim I/O areas. Using these selects and the eight address lines two 256

"... Understanding circuit operation is rewarding .."

byte I/O areas can be created – one from FC00 and one from FD00. The only other signal in the 1MHz bus worth a mention is the analogue input which is mixed with the output of the sound generator by a resistive mixer following IC17.

The Tube is another interface bus like the 1MHz bus but in this case it is not buffered in any way. A full copy of the data bus and lines A0 to A7 of the address bus is provided. In normal operation only 32 bytes of address space are allocated to the Tube but it is possible to disable a section of OS ROM (IC51) so that the Tube can make use of its full 256 byte addressing range.

The keyboard

The keyboard is interfaced to the rest of the system by the system VIA (IC3 but for most of the time it is 'free running'. In free running mode a four-bit counter (IC1k) is clocked at 1MHz and its output is applied to a one of sixteen selector (IC3k). In this way each column line of the keyboard is made active in turn. Any keypresses short a column line to a row line and this event is detected by IC4k which sends a signal to the system VIA which then causes an interrupt. Following a keyboard interrupt the free running mode is stopped and the keyboard handling software causes the processor to load column numbers directly into the four bit counter and row numbers directly into a one of eight selector (IC2k).

"... The BBC micro is still one of the most innovative micro computer designs ..."

In this way the the processor can test each key in turn to discover if it is pressed or not. (A pressed key will connect a row and column and when this combination is tested a pulse will be sent to the system VIA by IC4k).

Extra hardware

Three areas of the BBC Micro's circuit have not been described – the disk interface, the speech synthesiser and the Econet interface. Each of these is almost a complete system in itself and as such is beyond the scope of this introduction. However they are included in the circuit diagram and if you know how they are used you should have no trouble in following the way that they fit into the rest of the system.

All areas of the circuit that have been covered have only been done so very briefly. For a more detailed and leisurely discussion of the BBC Micro's hardware (and software) see my book "The BBC Micro: an expert guide" published by Granada (1983).

CENTRONICS WITH THE '64

Low cost and straightforward construction are the keynotes of this project.

We apologise to the author of this project – we have mislaid your name and address. Please get in touch with us as soon as possible. To all potential contributers, please print your name and address on ALL listings, diagrams etc submitted for publication.

The printer interface built into the Commodore 64 computer is a serial type designed to the RS232 standard. For the many CBM64 owners who own or have access to a printer with the more common Centronics parallel interface this is a cause of annoyance.

Fortunately for this group of '64 users the computer's user port can provide all of the signals that go to make up the Centronics control and data lines; it is only necessary to add a low cost interface and some software to meet the full Centronics standard.

Apart from the eight data output lines, a minimal Centronics interface must provide BUSY, ACKNOWLEDGE and DATA STROBE lines. BUSY is used to indicate to the computer that no data can be accepted, DATA STROBE loads the printer buffer with the byte on the data lines, while ACKNOWLEDGE is a negative pulse to indicate that a byte has been stored and that the printer is ready for another byte.

TABLE 1

CENTRONICS		COMMODORE	
PIN	FUNCTION	PIN	FUNCTION
1	DATA STROBE	B	PC2
2	DATA BIT 1	C	PB0
3	DATA BIT 2	D	PB1
4	DATA BIT 3	E	PB2
5	DATA BIT 4	F	PB3
6	DATA BIT 5	H	PB4
7	DATA BIT 6	I	PB5
8	DATA BIT 7	K	PB6
9	DATA BIT 8	L	PB7
10	ACKNOWLEDGE	B	FLAG
11	BUSY	M	PA2
19	0V	1	0V
—	—	2	5V

The user port of the CBM64 can provide all of these signals with the minimum of additional hardware and software. The 6526 PIA provides the required lines for handshaking: PC2 goes low for one cycle whenever the port is written to and so is ideal for DATA STROBE; FLAG is a negative edge sensitive input which sets a flag on the occurance of a negative edge and so is well suited to implementing the ACKNOWLEDGE line. The flag bit is bit four of the 6526's CRA located in CIA2. The only signal left to assign is the BUSY line from the printer, this is connected to bit 2 (PA2) of port A.

The only hardware requirement is to buffer all lines into and out of the computer. This is achieved by the use of two open collector drivers (IC1,2) and the printer provides the necessary pull up resistors on all lines. The only other requirement is that the very narrow output pulse from PC2 is stretched – achieved by the use of the monostable IC3.

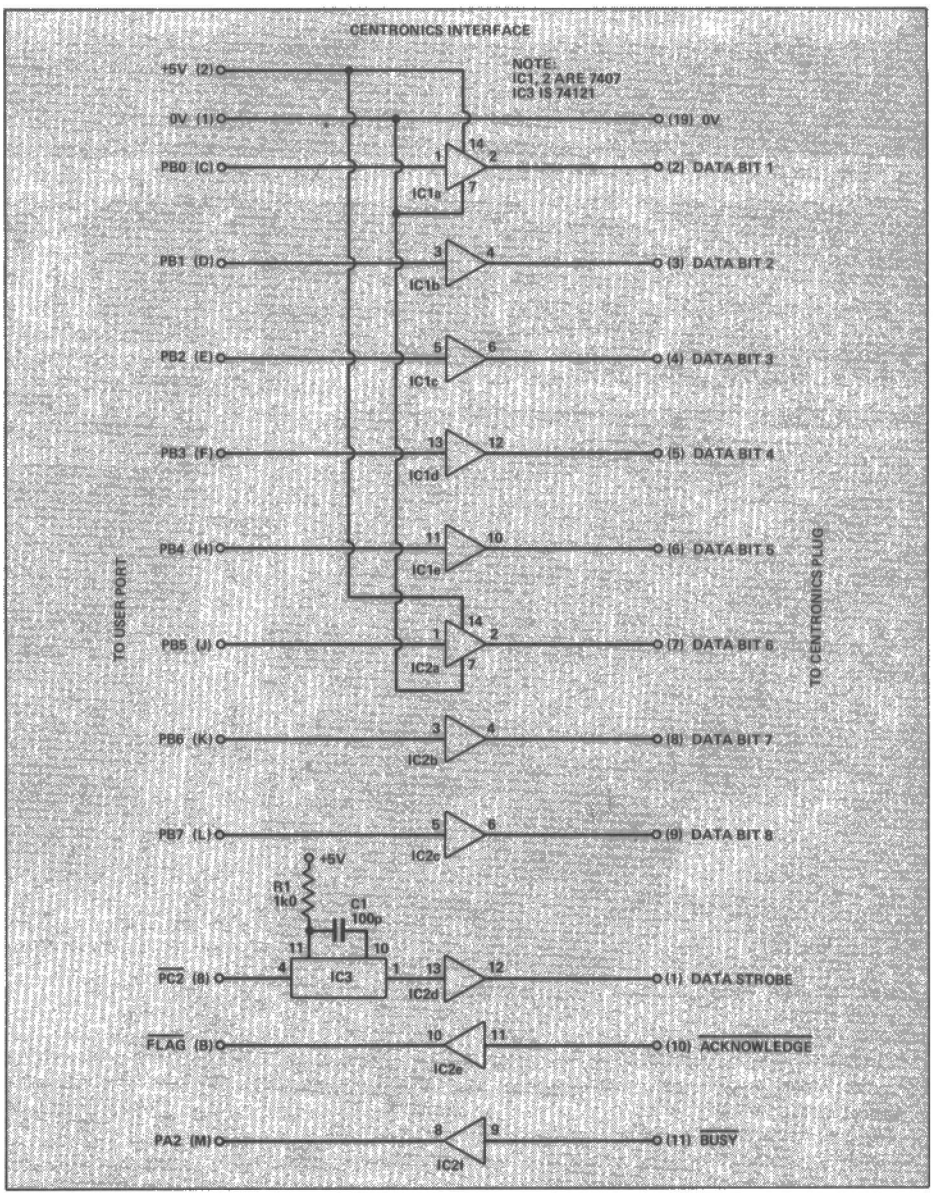


Figure 1. Circuit diagram of the interface.

LISTING 1

```
        PRINTER ROUTINE

.:C000 A9 FF 8D 03 DD A9 3B 8D
.:C008 02 DD 78 A9 18 8D 26 03
.:C010 A9 C0 8D 27 03 58 EA 60
.:C018 48 A5 A9 29 01 D0 04 68
.:C020 4C CA F1 AD 00 DD 29 04
.:C028 D0 F9 68 48 8D 01 DD AD
.:C030 0D DD 29 10 F0 F9 10 E7

 ENTRY
                LDA #$FF
                STA  $DD03 ; DDRA = ALL O/P
                LDA #$3B
                STA  $DD02 ; BIT 2 (BUSY) OF DDRB = I/P
                   SEI
                   LDA #$18  ;
                   STA $0327 ; NEW OUTPUT VECTOR
                   LDA #$C0  ; OLD=$F1CA:NEW=NEW.V
                   STA $0328 ;
                   CLI
                   NOP
                   RTS
            NEW.V PHA
                   LDA $A9
                   AND #$01
                   BNE PRINT ; PRINTER ENABLED
           EXIT    PLA
                   JMP $F1CA
          PRINT    LDA $DD00
                   AND #$04
                   BNE PRINT     ; PRINTER BUSY
                   PLA
                   PHA
                   STA $DD01     ; OUTPUT BYTE
                                   TO PRINTER
            ACK?   LDA $DD0D
                   AND #$29
                   BEQ  ACK?     ; BYTE NOT
                   BNE EXIT  ACKNOWLEDGED
```

The circuit diagram of the interface electronics is shown in **Figure 1.** As the circuit consists of only three ICs it wasn't necessary to design a PCB. Instead it's suggested that you build it up on a small piece of Veroboard.

Table 1 gives details of the CBM64's user port connections, for details of the connections to the printer refer to the manual supplied with the equipment.

After carefully checking the assembled interface it is time to test the project – for this it is necessary to enter the software.

The easiest way of driving the printer is to change the output vector at $0326. By changing the value held in this location it is possible to send to the printer all the output that would under normal circumstances go to the screen. A potential snag is that most users will not require that the printer is 'on' all the time. To overcome this problem, bit 1 of memory location 169 is used as a printer on/off toggle. With the bit set to 1 the printer is on and all screen output will instead be routed to the printer.

"... The interface works perfectly with Easy Script ..."

This may throw up yet another problem in that many printers will start to do strange things if some of the characters in the CBM's character set are not trapped. It is necessary to trap all non-printable characters such as the cursor control characters – users of the interface must pay attention to this point.

As shown the interface works with Easy Script without the need for the printer driver as the trapping of non-printable codes is already taken care of within this program.

All new in the 1985 Catalogue

From a gentle purr to a mighty roar, the tightly controlled power of the beast is yours to command!

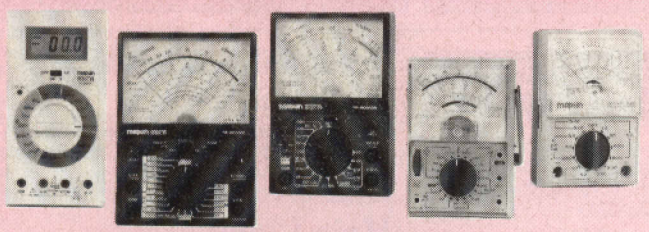
PROFESSIONAL QUALITY HIGH POWER LOUDSPEAKERS

A new range of superb quality loudspeakers.
- ***Virtually indestructible high temperature voice-coil reinforced with glass-fibre***
- ***100% heat overload tolerance***
- ***Advanced technology magnet system***
- ***Rigid cast alloy chassis***
- ***Linen or Plastiflex elastomer surrounds***
- ***5-year guarantee (in addition to statutory rights)***

Available in 5, 8, 10, 12, 15 and 18 inch models with 8Ω and some 16Ω impedances and with input powers ranging from 50W to 300W e.g.

5in. 50W 95dB 8Ω: XG39N / 16Ω: XG40T £17.95§
8in. 100W 98dB 8Ω: XG43W £29.95§
10in. 100W 100dB 8Ω: XG46A £29.95§
12in. 100W 101dB 8Ω: XG49D £29.95§
12in. Twin Cone 100W 100dB 8Ω: XG50E / 16Ω: XG51F £31.95§
Note - the output power doubles for each 3dB increase (ref 1W @ 1m).

PRECISION GOLD MULTIMETERS



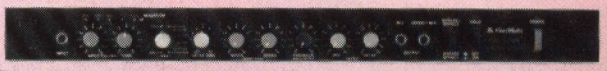
A new range of very high quality multimeters offering truly amazing quality at the price.

Pocket Multimeter, 16 ranges, 2000Ω/V DC/AC £6.95§ (YJ06G)
M-102BZ with Continuity buzzer, battery tester and 10A DC range, 23 ranges, 20,000Ω/V DC £14.95§ (YJ07H)
M-2020S with Transistor, Diode & LED tester and 10A DC range, 27 ranges 20,000Ω/V DC £19.95§ (YJ08J)
M-5050E Electronic Multimeter with very high impedance, FET input, 53 ranges including peak-to-peak AC, centre-zero and 12A AC/DC ranges £34.95§ (YJ09K)
M-5010 Digital Multimeter with 31 ranges including 20Ω and 20μA DC/AC FSD ranges, continuity buzzer, diode test, and gold-plated PCB for long-term reliability and consistent high accuracy (0.25% +1 digit DCV) £42.50§ (YJ10L)

Our huge range of top quality electronic components at very competitive prices are all detailed in our catalogue, and with well over 600 new lines in our 1985 edition and many design improvements, it's well worth getting a copy. Here are just a few examples from the catalogue. (The items below are NOT kits).

★ Most phono and jack plugs now with integral strain relief sleeve - gold-plated types also available from 14p (gold from 70p)
★ Stereo Disco Mixer with cross-fade, talk-over, cue monitoring, aux input, slide controls. Only £58.95 (AF99H)

★ 10-Channel Stereo Graphic Equalisers - 3 models - basic; with peak level meter; and with spectrum analyser - from £77.95



★ Digital Delay Line permits Slap-back, Doubling, Flanging, Chorus and Echo. 11 controls. Only £195.00 (AF98G)
★ Video Enhancer improves picture quality when recording from one VTR to another, and with TV's with monitor input. Only 28.95 (XG59P)
★ Detailed descriptions of the exciting new 74HC range of IC's which combine the advantages of CMOS and TTL. From 46p
★ Keyboards: sloping keys, two-tone grey, mounted in steel frame, very smart cases (extra) available. 61 keys, only £33.95 (YJ12N)
79 keys, only £37.95 (YJ13P)
★ 1% Resistors now 50ppm/°C, 0.4W, only 2p each!
★ Auto transformers 120/240V 50VA, £10.75§ (YJ56L). 100VA £14.95§ (YJ57M). 150VA £16.95§ (YJ58N). 250VA £21.95§ (YJ59P).
★ Digital Clinical Thermometer. Only £13.95 (FK51F)

Check our 1985 Catalogue for all our other fascinating new lines.

AMERICAN EXPRESS · map · Access · VISA

☎ ***Phone before 2pm for same day despatch.***

1985 CATALOGUE

Pick up a copy now at any branch of W.H. Smith or in one of our shops. The price is still just £1.35, or £1.75 by post from our Rayleigh address (quote CA02C).

N.B. All our prices include VAT and Carriage. A 50p handling charge must be added if your total order is less than £5 on mail order (except catalogue).

MAPLIN ELECTRONIC SUPPLIES LTD.

Mail Order: P.O. Box 3, Rayleigh, Essex SS6 8LR. Tel: Southend (0702) 552911
SHOPS
- BIRMINGHAM Lynton Square, Perry Barr, Tel: 021-356 7292.
- LONDON 159-161 King Street, Hammersmith, W6. Tel: 01-748 0926.
- MANCHESTER 8 Oxford Road, Tel: 061-236 0281.
- SOUTHAMPTON 46-48 Bevois Valley Road, Tel: 0703 25831.
- SOUTHEND 282-284 London Rd, Westcliff-on-Sea, Essex. Tel: 0702-554000

Shops closed all day Monday.

Post this coupon now for your copy of the 1985 catalogue. Price £1.35 + 40p post and packing. If you live outside the U.K. send £2.40 or 11 International Reply Coupons. I enclose £1.75.

Name .....
Address .....
.....
.....

ECM05

§ Indicates that a lower price is available in our shops.

All offers subject to availability.

Prices firm until Aug 10 1985.